NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

LINKING INFORMATION FOR MOBILE USE

by

Robert N. Myers
Edwar Zapata

September 2007

Thesis Advisor:	Gurminder Singh
Co-Advisor:	Arijit Das

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2007	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Linking Information for Mobile Use			**5. FUNDING NUMBERS**
6. AUTHOR(S) Robert N. Myers Edwar Zapata			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**
13. ABSTRACT (maximum 200 words) Information Management (IM) has been an area of research and discussion for several decades. Studies have been conducted by behavioral and computer scientists on how people organize their information and workspaces in order to come up with efficient ways to store, organize and retrieve information on personal computers. This thesis explores improving a user's ability to manage information on mobile devices. The goal is to make placing information on such devices a more attractive prospect, with an emphasis on retrieval of stored information regardless of the document type. This will result in mobile users having quick access to the right information at the right time while away from the office or home. This thesis describes the challenges inherent in a mobile scenario and the system designed to address those challenges. The system provides visual and navigational features that are not currently available on mobile devices, specifically the ability to view multiple types of items in a single interface. Additionally, the ability to logically link related items as an IM tool is examined.			
14. SUBJECT TERMS Mobility, Personal Information Management, User Interface, Interactive Information Retrieval, Groups, Linking.			**15. NUMBER OF PAGES** 95
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

LINKING INFORMATION FOR MOBILE USE

Robert N. Myers
Captain, United States Marine Corps
B.S., St. Peter's College, 1999

Edwar Zapata
Captain, United States Marine Corps
B.S., University of Illinois at Chicago, 2001

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2007

Author: Robert N. Myers
Edwar Zapata

Approved by: Gurminder Singh
Thesis Advisor

Arijit Das
Co-Advisor

Peter J. Denning
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Information Management (IM) has been an area of research and discussion for several decades. Studies have been conducted by behavioral and computer scientists on how people organize their information and workspaces in order to come up with efficient ways to store, organize and retrieve information on personal computers. This thesis explores improving a user's ability to manage information on mobile devices. The goal is to make placing information on such devices a more attractive prospect, with an emphasis on retrieval of stored information regardless of the document type. This will result in mobile users having quick access to the right information at the right time while away from the office or home. This thesis describes the challenges inherent in a mobile scenario and the system designed to address those challenges. The system provides visual and navigational features that are not currently available on mobile devices, specifically the ability to view multiple types of items in a single interface. Additionally, the ability to logically link related items as an IM tool is examined.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

Robert Myers:

I would like to thank my sister Barbara for being a great friend and my most die hard supporter.

Edwar Zapata:

I would like to thank my wife, Pamela, for all of her support throughout my entire military career. Her dedication to our relationship and family has helped me get to this point in my career. I also thank my two boys, Nikolas and Christian, for being the great kids that they are, and for being able to adjust as well as they have to the military lifestyle that they are forced to live. Last, but definitely not least, I thank my thesis partner Rob Myers for all of his hard work in developing the application. It has been a privilege working with him, and I look forward to possibly working with him in the future.

We would both like to thank Professor Gurminder Singh and Arijit Das for their efforts in helping us develop the MPM system and this thesis document.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. INFORMATION MANAGEMENT

The challenges inherent to managing information, personal and professional, have not changed in a long time [1]. First, work must be done during the acquisition of items to form a collection. Second, the collection must be organized. Third, the collection must be maintained for long term storage. Finally, items must be retrieved for use. These activities form the core of any information management strategy, starting with paper and pencil and continuing to the multitude of networked, digital, handheld devices that are currently available. In 1945, Vannevar Bush [2] stated, “A record, if it is to be useful to science, must be continually extended, it must be stored, and above all it must be consulted.” Doing this well is still a goal that most people struggle with every day, regardless of the type of information being managed, and regardless of the type of device being used to store and manipulate the information.

These days, the amount of information a person must manage grows rapidly. In addition to the shear quantity of data, the number of places it is located is also growing as mobile devices such as cellular phones and personal digital assistants (PDAs) become more capable and commonplace. At the moment, many users are hesitant to manage information on their mobile devices. They are dipping their toes in to feel the water’s temperature, but few are diving in to place a more substantial amount of information on handheld devices for a myriad of reasons. The two goals of this thesis research are to identify the usability reasons that stand in the way of using the full capabilities of today’s mobile devices, and to design and present a system that makes managing information on a mobile device simpler and potentially more attractive to the reluctant user.

B. MOTIVATION

When discussing IM, it is easy to forget that the goal is not just to sort or search better, or to make the interface more appealing. The goal is to help the user accomplish some task more efficiently, such as planning a trip, or managing an event. Performing such tasks on a hand-held device is still not an appealing idea to most users, so the expected efficiency gains from being able to accomplish IM tasks away from the office are not being seen. Getting the raw data on the device is no longer the challenge; easily using the data once you have it on the device is.

The motivation for improving a person's ability to use acquired information is twofold. First, the problem of information overload still exists in the desktop/laptop world, so one cannot simply extend practices from that world to the mobile one. Just as extending paper and pencil models to the digital world has been insufficient, extending desktop file management to handheld devices is also inadequate. New interface models are needed. Second, the correct information is not as good as the correct information at the correct time. The correct time for a mobile user is most likely while he or she is away from the office, so it is highly desirable to encourage and facilitate placing and using all needed data on a user's mobile device.

C. MOBILE DEVICE SPECIFIC CHALLENGES

When developing applications for mobile devices, there are certain mobile device specific challenges that the programmer has to keep in mind. The first and possibly the most important is the power limitation. Due to the mobile nature of the device, it is usually not connected to an external power source. Therefore, the application should try to make use of the device's power saving features. Another challenge that the developer has to deal with is the display size. This problem is actually two-fold because the system developer has to take into consideration the variety of screen sizes available on the various hand-held devices and he has to be able to use the limited amount of screen real estate to display the necessary information clearly and in meaningful ways. Memory consumption is a third concern related to mobile device programming. Unless a person is using an external storage device such as a storage card, mobile devices currently have limited storage capacity. The system code therefore has to be more efficient in comparison to code developed to run on a PC or laptop. Another limitation that must be addressed is the processor speed. Despite tremendous progress in processor technology, processors on hand-held devices run significantly slower than those on PCs and laptops. For this reason, a great deal of thought must be placed into choosing the appropriate data structures, and into keeping all algorithms as efficient as possible. Finally, input modalities for mobile devices are different; mouse and keyboard interaction are replaced with stylus clicks and soft buttons. Planning for this type of user interaction is different and requires a thorough analysis of user interaction with the software system and the device.

D. RESEARCH QUESTIONS

Our research focused on five major aspects of personal information management on mobile devices:

1. Optimal ways to store and organize information that is imported or recorded to the device. .

2. Identification of what stored information is most useful.

3. Efficient information retrieval and use.

4. Automating device actions that can improve the user's ability to manage their schedule, projects, and personal activities

5. User interface design to optimize the small amount of screen real estate available on mobile hand-held devices, while simultaneously being simple to operate.

We examine a wide spectrum of research in the area of information management dating as far back as 1945. Our goal is to extract core IM concepts that have been identified from then to now and apply what is most appropriate to

IM on handheld devices. We use the research data to design a personal information management system for mobile hand-held devices that run the Windows Mobile 5.0 operating system. Upon completion of the prototype, we gathered usability data in order to determine the effectiveness of the system.

E. THESIS OBJECTIVE

The overall objective of this research is to identify universal information management behaviors that are effective and transcend the tools that have been developed over the decades. Once these behaviors are brought to light, tools can be developed to accommodate and enhance them. Based on personal observation and the research conducted, we have designed an application that would make a user's information easier to access and retrieve, and would automate the organization of this information once it is placed on the mobile device. We improved accessibility by designing an interface that makes all information on the device viewable in a single location via a simple tabbed browsing layout. Automating the organization of the information is accomplished in the following manner. First, related items are automatically grouped and can be viewed on the same screen rather than opening one application to view e-mail, another to view SMS messages, and yet another to view appointments. Second, items such as e-mail messages frequently imply the creation of new items such as an appointment or a to-do list. Creating these additional items can be difficult and frustrating on a mobile device due to the limited input modalities, so they are frequently not created. This results in the information in the original item being lost in the e-mail or SMS, making later retrieval difficult. The attempted solution for this issue is to automatically create these supporting items using the information contained in the root item, such as an SMS message. A person will be more willing to load her mobile device with information if the device becomes an organizer and in some cases a creator of information rather than a black hole for received information.

F. THESIS ORGANIZATION

The thesis is organized into six chapters. Chapter II highlights the key papers that identify the core IM principles that have been revealed over the years. Chapter III describes the application we call Mobile PIM Master (MPM) that is capable of managing most of the data types that can be found on current handheld devices through a single integrated interface. Chapter IV describes the software design and development of the application. Chapter V presents the evaluation data that was gathered based on usability and performance tests. Finally, chapter VI details the conclusions and recommendations for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND OF PIM RESEARCH

A. INTRODUCTION

Throughout the course of the past six decades both behavioral scientists and computer scientists have conducted studies on the various ways that different people organize their information and workspaces in order to come up with an efficient way to store, organize, and retrieve information on personal computers. As early as the 1940s, researchers began to see the need for developing a method for storing, organizing and most importantly retrieving captured information. As technologies such as still photography, television, teletype and radio continued to flourish, the need to efficiently store the growing amount of data captured by those technologies was identified. Since then, the number of data types such as text documents, pictures, emails, audio files, etc. has increased along with the amount of information captured in those data collections. With that growth, comes the need for more efficient ways of managing the large volumes of information. Early research in this area has focused on one type of information, such as e-mail, appointment scheduling, or alerts. Some research within the past decade has begun to highlight the need for more information integration tools. A synthesis of this research is attempted to extract the best practices identified in each of the following papers.

The research is presented in chronological order to show how the field of Personal Information Management has developed over the last sixty years.

B. THE “MEMEX”

In July of 1945, Vannevar Bush, an American engineer and science administrator, published an article, “As We May Think,” in The Atlantic Monthly [2]. In his article, Dr. Bush acknowledged some of the technological advances that had been made at the time, made some predictions about the future growth and implementations of those technologies, and identified a need for what he called a “memex,” “a device in which an individual stores all his books, records, and communication, and which is mechanized so that it may be consulted with exceeding speed an flexibility.”

The article placed emphasis on those technologies which allowed people to capture information about their daily findings and activities such as photography, voice recorders, typesetting, and printing. He provided a scenario wherein by combining the technologies, a researcher had the ability to observe, comment and capture data without being anchored to a desk or table. Dr. Bush saw that with all of the available capabilities to collect data and observations came the need to store, organize and retrieve those collections. In his words, “A record, if it is to be useful to science, must be continuously extended, it must be stored, and above all it must be consulted.”

He further went on to discuss the need for a systematic approach for storing the information, one that would be intuitive and could possibly mimic the way that the human mind organizes information. He proposed that instead of using an alphabetical or numerical filing index, an organization based on association of related items would prove more useful in the retrieval of the stored information: “The process of tying two [related] items together is the important thing.” Dr. Bush’s article was insightful and it provided the foundation for further research in the area of information management.

C. ”WHERE IN THE WORLD IS THE INFORMATION?”

In 1968, George A. Miller, psychology professor and founder of WordNet, a linguistic knowledgebase that maps the way the mind stores and uses language, focused his attention on the area of Information Management. In July 1968 he published an article, “Psychology and Information,” in the American Documentation journal [3]. The article focused on the way that computer-based systems stored information and the methods used to retrieve the information. His observation was that the designers of the systems were not taking into consideration the process by which the human mind stores, categorizes, and retrieves information, and in so doing, information retrieval was not being done effectively.

The two key arguments of the article were that people are used to locating information spatially, and that they are able to retrieve information more effectively if it is presented in a format that they are can relate to, such as an organized library book shelf in which books of the same topic are located in the same place. The author supported his theory of spatial locality by presenting some examples of how people were able to recall information better by associating the information to spatial position. One example was that of a teacher trying to remember the names of his students. He noted from personal experience and discussions with colleagues that it was easier to remember the names of new students if they sat in the same seats for the first few days of class, than if they changed seats.

Another example was an experiment that was conducted with a chimpanzee. The chimpanzee was shown two boxes that are easily distinguishable by size and color. One box was larger than the other and was painted white. The smaller one was painted black. The large white box was placed on the left, and food was placed inside. A curtain was then drawn and the location of the boxes was changed. After the switch was made, the chimpanzee was allowed to see the boxes again, and reach for the food. The observed behavior was that the chimpanzee reached toward the location of where the food had been placed. Both examples showed the need for further exploration of the use of spatial locality as it relates to information management in computer systems.

Miller supported his second argument that users could retrieve information more effectively if the information was presented in a familiar format, with the use of a library analogy. Librarians place books of related content in the same vicinity. When patrons are in search of books related to a particular topic of interest, they are able find the specific books that they are in search of, and they can easily see books that are related to the topic as well. New books and journals are also placed in an area that is easily noticed in order to provide the patron with easy access to the most current information. Additionally, the books are clearly labeled and a precise indexing system is established in order to facilitate retrieval of information. This method of organization provides both a simple method for locating exactly what you're looking for as well as a visual reference that aids the patron in locating related material.

Miller addressed two key weaknesses of information management in computer systems, and he identified some possible solutions to strengthen those weaknesses. He concluded his paper with a message that IM tool developers should keep in mind; "Mankind evolved in a world of space and time. Our memories evolved to record events that transpire in space and time. Modern attempts to externalize and enlarge that memory should not and probably need not, neglect its spatial temporal locality."

D. "FILES" AND "PILES"

Thomas W. Malone, Professor of Management at the MIT Sloan School of Management, conducted a study, and wrote an article in 1983, "How Do People Organize Their Desks? Implications for the Design of Office Information Systems," that focused as the title states, on how different people organize their workspace [4]. The article made two principle claims. The first was that along with helping a user find desired information, reminding is also an important function of desk organization. The second claim was that computer-based systems could assist with some of the cognitive difficulties of organizing information by "(a) doing as much automatic classification as possible (e.g., based on access dates) and (b) by including untitled "piles" of information arranged by physical location as well as explicitly titled and logically arranged "files." Malone supports his claims by interviewing ten people on how they organize their desks and office. Based on the results of the interviews, he made some suggestions for the design of computer-based information systems.

The participants were divided into two categories; “Neat” office and “Messy” office. The “Neat” office group used a filing system and organized piles. The “Messy” group relied mostly on loosely scattered piles. Malone determined the separation of the groups by touring their offices and observing their organizational techniques.

The paper discussed two specific cases, each representing the two methods of office organization. The first case examined the organizational process of a member of the “Neat” group, Michael. Michael was a purchasing agent whose work primarily involved the use of standard forms. The office was arranged in a format that supported the paper flow. He had incoming and outgoing document trays, labeled piles of in-progress forms, a log book to document purchase order numbers, and file drawers arranged by subject. The second case focused on the organizational process of a member of the “Messy” group, Kenneth. Kenneth was a research scientist with little routine paper flow. Most of the information in the office consisted of magazines, books, papers, personal notes, and computer listings. Unlike Michael’s office, Kenneth’s office was filled with various loosely stacked piles of mixed content. Documents on the desk were not arranged in any distinguishable order other than to Kenneth, and the floor and tables were cluttered with stacks of books, magazines and additional papers.

The results of the tours and interviews showed that the two main units of desk organization were “files” and “piles,” and that regardless of the organization technique, the two main functions served are “finding” and “reminding.” As Malone pointed out, “The distinction between finding and reminding rests on intentionality.” Finding occurs when a person knows what he is looking for, where as reminding occurs when a person becomes aware of something else unintentionally. In most cases, a well labeled filing system helps in the “finding” process because a person can easily locate specific items that he or she has in mind. “Piles;” however, because of their location, whether they are labeled or not, serve to remind a person of unaccomplished tasks.

Based on the analysis of the interviews, Malone pointed out that computers can be used to simplify the three processes required to find information; “creating classifications,” “classifying information,” and “retrieving information,” by allowing for multiple classifications, by automating classifications, by deferring classification, and by allowing the user to retrieve information using more than one dimension. In regards to reminding, Malone suggested four possible priority schemes to aid the user: Frequency, size, location, and color. In the frequency scheme, higher priority items are displayed more often. The size scheme would use larger icons for higher priority items. Higher priority items in the location scheme would be displayed in specific areas of the computer screen. The color scheme would use different hues to signify the level of priority.

Malone’s study brings to surface again the concept of spatial locality, and he mentions the importance of incorporating it into computer-based systems. Though his study focused mainly on the organization of paper documents, a great deal can be learned about information management from his study of human behavior regarding organization of information.

E. PIM IN THE 1990S

1. Finding and Reminding

In 1995, Deborah Barreau and Bonnie A. Nardi collaborated to write an article [5] about their independent studies on the ways users organize and find files on their computers. Their motivation for conducting the studies was two-fold. First, since filing and finding are such integral parts of computer usage, more research needed to be conducted in order to try to optimize the processes. Second, along with the surge of networking capabilities, a growth in information storage and sources emerged so new methods needed to be developed to manage the additional resources. Barreau examined information management techniques practiced by DOS, Windows and OS/2 users. Nardi focused on filing and finding methods implemented by Macintosh users. After having conducted their research, they compared their analyses and determined that there were more similarities than differences. Some details about the studies are presented in the following paragraphs along with the findings, and the recommendations for improvement provided by the authors.

The authors conducted their studies at slightly different times, and unbeknownst to each other. They decided to collaborate on the article after they found out about each other's work. Barreau conducted her study in 1993. She focused her attention on how seven office managers organized and retrieved information from their electronic system. The level of computer experience ranged between novice and proficient. Four of the managers used DOS, one used Macintosh, another used OS/2 and the final manager used Windows 3.0, and they were all connected to servers. The goal of her research was "to identify the types of documents used and to determine the factors affecting individual decisions to acquire, organize, maintain and receive information." In 1994, Nardi interviewed fifteen Macintosh users as a basis for her research. The participants included administrative assistants, programmers, graphic artists, manager and librarians, and were all experienced users. The users were asked to provide a tour of their system and were asked a structured set of questions in order to identify individual approaches to organizing and filing folders.

After comparing their results, they found four similarities and only one difference between the two study groups. Three of the similarities were of particular interest for our research. The first similarity was the preference for location-based search to find files. The two basic strategies for finding files are location based and logical. In location-based finding, the user guesses at the location of where she thinks the object of interest might be and conducts a visual search for the object. If she doesn't find it, she repeats the process. In logical finding, the user finds files using a text-based search. This requires the user to remember the name of the object she is looking for. Both studies showed that location-based search was by far the more appealing method.

The second similarity was the reminding function of file placement. In both studies, participants placed files in locations where they were most likely to notice them such as the upper level of a directory or certain areas of their screen. The reminding function ties in to the preference for location-based search. As the authors pointed out, since file placement serves both the finding and reminding functions, “a ‘physical’ system in which a specific location is associated with the file is more useful than a purely logical system.”

A third similarity that they found was that each group’s information was divided into three categories: “ephemeral,” “working” and “archived.” Ephemeral information has a short life and includes items such as tasks, e-mails, sms, appointments and notes. Working information are items that are often created by the user and are frequently used. It is usually put in its own location. Contacts would fall into the working information category. Because the information has a prescribed location and is frequently accessed, users have little issue finding the desired information. Archived information has a long shelf life, is not used frequently and in most cases is indirectly related to on going work. This type of information proved to be the most difficult to organize across both groups because according to the participants, the effort required to do it was not worth it. Of the three types of information, ephemeral was the most difficult to manage across both groups because of the volume and rapid fluctuation.

The authors discovered that it is important to consider the various types of information when designing a filing and finding system. In their opinion, ephemeral information organization had not received enough attention, while archived information, which is accessed and required the least as their study showed, was over emphasized. In an era when mobile devices are becoming so prevalent and information storage is so distributed it is even more important to develop more effective ways of organizing information so that it serves the two main purposes identified by Nardi and Barreau: *finding* and *reminding*.

2. Sorting out Searching

In the April 1998 edition of Communication of the ACM, Ben Shneiderman, Donald Byrd, and W. Bruce Croft presented an article [6] that focused on the issues of information sorting and searching as it relates to personal and distributed databases. The authors highlighted several user-interface design flaws such as inconsistencies among different systems and the lack of suitable explanations regarding the methods used for conducting the searches. They pointed out that many times users have a hard time conducting productive searches because most text-based search systems do not provide either clear instructions as to how to input search words (i.e., word combinations and use of operators), feedback or suggestions for refining the search. In an attempt to alleviate some of those issues, the authors proposed a four-phase framework for developing a well designed user-interface that provides both a common structure and ease of navigation.

The goal of their framework was to "increase clarity and user control while reducing inconsistencies in text-search user interfaces." The four phases consist of a formulation phase, an action phase, a review of results phase and a refinement phase. The **formulation** phase is the first phase and the most complex because it deals with cognitive processing, what text to search for, the different variants of the text, where to search and which fields of the location to search. The **action** phase is the phase in which the search is actually conducted. For implementation of this phase, the authors recommend the use of "dynamic queries" as opposed to just clicking on a Search button to initiate a search and waiting for the results. With this technique, the result set is continuously modified and displayed as the user changes the search. **Review of results** is the third phase and it deals with how the results of the search are displayed. The authors identified various techniques for displaying the information and suggested that the interface should also provide messages to aid in refining the search. **Refinement** is the fourth and final phase. This phase implements both word relevance feedback and a search history buffer. The history buffer assists the user in conducting more productive searches and the buffer allows for review, alteration and resubmission of previous searches.

The authors pointed out some points of weakness in the area of text-based search, and they presented some possible solutions to strengthen the process. They concluded by reiterating the importance for developers to continue their work in refining search interfaces. They also stated that the future of the Web was dependent on the developer's abilities to simplify the search process for the masses of users. Though they focused mainly on Web queries, their concepts can be applied to various PIM tools.

F. PIM IN THE NEW MILLENIUM

1. "Stuff I've Seen"

At the 2003 Special Interest Group on Information Retrieval (SIGIR) conference, Susan Dumais et al. presented an article [7] that described the design and evaluation of a desktop system that they called *Stuff I've Seen* (SIS). The system provides two main features. First, it maintains a unified index of previously seen information such as emails, web pages, documents, etc. Second, the search interface uses "rich contextual cues" to assist in the recognition and retrieval of items. The goal of the system was facilitate finding and re-using of information that has been seen before by the user.

The system architecture is comprised of five main components. The **Gatherer** provides the interface between the different content sources and the system. The **Filter** decodes the various file formats (i.e., pdf, html, doc) and produces a character stream. The **Tokenizer** breaks the character stream into "words." The **Indexer** builds the index structure with position information to promote rapid retrieval. The **Retriever** is "the query language for accessing stored information." The five components work together to provide the local unified index, and the index is updated continuously as new items are received or created.

The SIS interface provides the user with the ability to initiate queries and to view and manipulate the results. It is predominantly a text-based search display, but it does provide some visual cues. Five filter fields are displayed in the view, and queries occur dynamically so they are launched whenever any of the filter boxes in the user interface are manipulated. Query results are initially sorted by Date or Rank, but selecting a column will sort the information by that column.

The authors presented a system that took into consideration the various PIM items that people deal with on a daily basis, and developed a desktop based system that integrates the various information types and displays them in a consolidated view. They planned to continue development of the system. Future work plans included improvement of the search engine and exploring different visual presentation of results. They also mentioned adding additional features such as context awareness..

2. “Stuff Goes into the Computer and Doesn’t Come Out”

In April 2004, Boardman and Sasse, presented their paper, “‘Stuff Goes into the Computer and Doesn’t Come Out’ A cross-tool Study of Personal Information Management,” at the Conference for Human-computer Interaction [8]. The paper reported a study of Personal Information Management with two main areas of interest. The first focused on cross-tool data involving files, email and web bookmarks. The second involved the collection of longitudinal data for a subset of participants in order to observe how PIM behavior evolves over time. Their objective was to improve “the empirical foundation for PIM design.” The following paragraphs will provide a brief description of the authors’ research, highlight the key results, and present their suggestions for improving information management.

The authors began their study by identifying some significant shortcomings of previous research in two areas; empirical studies and prototype design. In the area of empirical studies, they noted two significant limitations. First, they pointed out that little consideration had been given to PIM as a “cross-tool activity," and they brought two questions to light: "Do individuals employ similar strategies in email as in files? How are PIM tools used together?" Second, they identified that little attention had been focused on the evolution of PIM strategies. Regarding prototype design, they drew attention to the inconsistencies between what had been learned from tool-specific studies that had been conducted and what was being applied to the design of cross-tool integration.

The study consisted of two phases; *Profiling PIM Practices* and *Longitudinal Tracking of PIM Practice*. During the first phase, the authors interviewed 31 participants and requested a guided tour of their file, email and bookmark collections. They performed a content analysis of the data to determine "strategies," "problems" and "needs." They also analyzed folder structure to examine naming conventions and identify the number of folders sharing common collections. Finally, they compared the different organization strategies to determine if there were any behavior similarities.

The second phase focused on the evolution of the three collections and the management strategies. Only eight participants from Phase 1 took part in the second phase. The author's developed a tool to capture snapshots of the folder structures and their contents. They also asked the participants to maintain a log of significant events such as creating a new folder of failing to locate an item. Phase 2 also involved using WorkspaceMirror (WM), a software prototype developed by authors as a result of the folder overlap observations in Phase 1 [0]. The authors describe WM as an application that "allows the user to mirror structural changes (i.e., the creation, deletion, renaming or moving of folders) between the file, email and bookmark folder hierarchies."

The results of the study showed that the various participants used multiple PIM strategies within specific collections. Also, the cross-tool data showed that PIM tactics varied considerably between tools for many individuals. The observation of folder overlap brought to light a subset of user activities that involved PIM across the various information types. The information gathered about overlapping folders suggested that information concerning "*roles*" and "*projects*" may be usefully shared between collections.

The authors acknowledged the importance of PIM-tool integration design studies. They stated that "Cross-tool studies can provide an empirical foundation for such design by highlighting: (1) synergies between tools that can be exploited to improve integration and, (2) differences between tool usage that may indicate barriers to integration." They also point out that designers need to take into consideration the differences between individual users and the users' various PIM strategies. Finally, the authors identified the possible need for integrating files and filed emails, and they indicate that their future work plans would involve integration of other PIM tools and devices.

G. ADDITIONAL RESEARCH

Chirita et al [9] explored a way of clustering related items based on the number of steps between consecutive file accesses and the length of the time window between the access occurrences. This clustering is very similar to the grouping concept, but it uses temporal locality rather than content similarity to define the relationship among the items. Bellotti et al [10] worked with the idea of the task vista, a comprehensive list used for planning, on both the desktop and PDAs. The focus of this work is on improving a user's ability to perform task management. A personal task list manager called TaskVista was designed and implemented to allow simple creation of tasks, and the dragging and dropping of related items (e-mails and files) into the task. Bellotti and Smith [11] also identified the utility of the concept of groups that can be treated more like piles of needed resources than traditional folders. The use of spatial or temporal locality in combination with related content shows promise for creating accurate "groups" of information. Expansion of this grouping research is especially warranted on handheld devices.

Cignini et al [12] address to concept of automatic categorization of information. They designed a filtering, categorizing, and alerting e-mail system that processes incoming e-mail and performs automated classification operations. The objective is to spare the user from having to manually filter and sort e-mail while on the go. Such automatic categorization is a little too processor intensive for a mobile device, but a simpler form of automatic grouping can be explored.

Interface design or presentation of retrieved information is another element of information management that has attracted attention. Dontcheva et al [13] focused their efforts on web content and designed an interface that allows a user to select the elements of a webpage that are of interest, automatically collects similar content, and presents all of this data in a visual summary appropriate for a small screen device.

Work at Microsoft by Cutrell et al [14] on an application called Phlat attempts to improve a user's ability to search and organize personal data. This is a desktop application, but some of its key features can also serve a mobile user well. The features include the ability to view content from various storage systems, such as e-mail, files, or schedules in a single interface, as well as a querying capability that indirectly links several seemingly disparate items into a grouped query result that can be saved to maintain the grouping.

Additional work done at Microsoft by Czerwinski [15] focuses on improving a user's ability to multitask. Two applications, GroupBar and Scalable Fabric, are described. The GroupBar application allows a user to group or regroup related windows for easier screen organization, but no persistent logical links are established. The applications developed were for the desktop environment, but the need for effective tools is even greater in a mobile scenario where the user must divide her attention between several items on the device and activity in the real world, like not walking into traffic. The concepts presented in all three papers have yet to be rigorously applied to information management on mobile and handheld devices.

In the realm of mobility, Singh [16] explores the idea context awareness as it relates to delivering of personal information, specifically phone calls, emails and SMS, in appropriate forms depending on the user's context. For example, important emails can be delivered to the user as SMSs if the user is in a meeting. The idea is that while the user is in a meeting, SMSs can be delivered in a "silent" mode where the information can be delivered and consumed without disturbing the meeting. The same email is delivered as a phone call or voicemail, if the user is driving. SMS in this context will not work as the user will not be able to read SMS while driving. He may, however, be able to take the phone call.

Shimakawa, Kaede, Nakamura and Azuma [17] also investigate the concept of context awareness. Their paper discusses a method for interleaving independent daily tasks into an efficient flow of actions. The authors consider each task as a thread and evaluate the context of each thread to reorganize the independent sequence of events into a single efficient sequence. The paper discusses using data modeling for states of a user and objects around him, contexts the user has, and associations between items in the schedule to make efficient schedules. The authors propose three things to accomplish tasks using context. First, the functions of a PIM tool need to be classified in three layers: facts, contexts and planning. Second, set stereotyped actions so users do not fail to carry out incidental acts because they are only paying attention only to essential acts. Finally, introduce pre-conditions and post states for each act so users can insert an tasks from a to-do list into a schedule without inconsistency..

Akbas and Singh present a system for locating personal information across multiple mobile devices and web resources [18]. They reiterate the increasing amount of storage banks available to the common user (i.e., smartphones, laptops, PCs and web storage), and they also highlight the ever growing amount of computational power available on mobile devices. Their system allows users to initiate queries from any of four platforms: phone, laptop, PC, internet service, and it search all other devices and platforms that it is associated with.

H. THE “MESS”

Peter J. Denning [19] recently wrote an article, “Mastering the Mess,” that looks into the innovative technologies that have been developed to deal with historical *messes* and provides insight into how to identify current *messes* so that new innovations can be developed to deal with them. In the article, Denning provides examples of previous *messes* and identifies several current *messes*, one of them being information overflow. This mess persists because no comprehensive IM solutions have emerged from the multitude of research, but the research does identify the key ideas that must be explored. These ideas include linking related items, exploiting spatial locality, and using text-based search as a backup “finding” method. Groups or “piles” have also been identified as excellent tools to augment traditional file structures, and finally, integrating PIM tools is needed to deal with truly complex IM tasks. Building on the research that has been discussed and dialogue with peers, we have developed a system that attempts to tackle the information overload mess on handheld devices. Follow-on chapters describe the design of our system and the development process.

III. MOBILE PIM MASTER

A. OVERVIEW

The research and test implementations that we have examined demonstrate that a mobile user can get more value out of using a mobile device if a mobile IM application achieves the following objectives. First, the system must have an integrated interface that allows a user to view all pertinent items, regardless of type. This IM requirement is challenging because of the small amount of screen real estate available on a mobile device. More information is not always better on a small screen, but navigating among multiple applications on a handheld device can be far more frustrating. Second, creating logical connections between related items and clearly visualizing these connections can enhance a user's ability to retrieve and manage various tasks, and help the mobile multi-tasker to easily access information on the device while engaged in other activities. The concept of a group [10], or pile, has a great deal of potential as an alternative file organization method on handheld devices because navigating through multiple file structures is cumbersome on these devices. A system should allow the user to create groups manually, as well as provide some tools that create groups automatically.

In order to accomplish these objectives, we have developed a system, called Mobile PIM Master (MPM) that implements a single interface for navigating through all device content without overwhelming the user or the device. The system also implements a logical linking capability that allows the user to create groups containing any type of item from anywhere on the device. First, we designed the interface to be a comprehensive navigation area that can access all data types in a single location without overloading the screen. Second, we adopted the concept of a "group" [10] for logically linking related items and visualizing the relationships for the user. Third, we implemented a number of automatic features for simplifying item creation and grouping. Finally, we built a search tool into the system, allowing the user to search the device without having to start a new application. The user can add the search results to any group that he or she creates.

B. IMPLEMENTATION PLATFORM

We developed MPM for the Microsoft Windows Mobile 5.0 Pocket PC Phone Edition platform. We choose this platform because of the wealth of classes that are available in .NET Compact Framework for accessing the various information stores on the mobile device. Additionally, the capabilities of this platform match the capabilities of the most common smartphones and PDAs. Identification of the need for better IM tools on these devices drove our selection of the platform. The release of iPhone from Apple demonstrates the increasing availability and popularity of devices that have capabilities beyond that of a simple cellular phone, increasing the need to study IM on smartphones.

C. INTEGRATED INTERFACE

MPM maximizes the use of the limited screen real estate, making it possible to create a central interface that allows a user to view and manage the various information stores on the device including Contacts, Tasks, Appointments, E-mail, all files, any saved URLs, and all the groups created by the user. Additionally, the user can launch a search query directly from the MPM interface. A user can view, edit, or delete any of this information, and new items can be created without having to start a new application.

Rather than having the user open multiple applications and continuously activate and minimize different application screens, it is easier to allow the user to navigate through his data via a single interface. Figure 1 presents our interface designs. We developed two interface layouts. The first layout has a navigation bar on the left side of the screen and the second has icons along the bottom of the screen that serve the same function as the left navigation bar. The user can toggle between the two views depending on whether quick navigation is the priority, or maximum viewable screen area is the priority. All data (e.g. files, contacts, appointments, e-mails, etc) on the device are accessible from this interface, as well as the groups created by the user. Figure 2 displays the options for the user if he or she wants to change views.

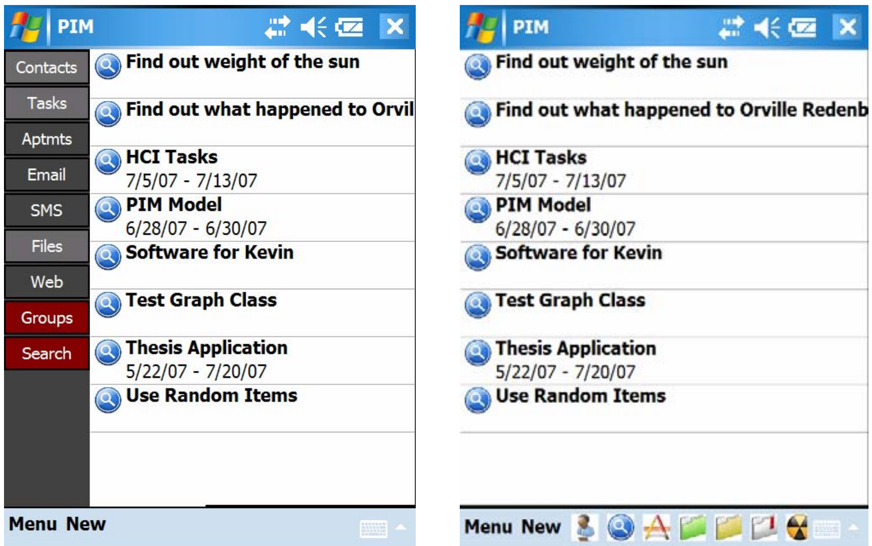

Figure 1. (Left) Interface with left navigation bar. (Right) Interface with lower navigation bar.

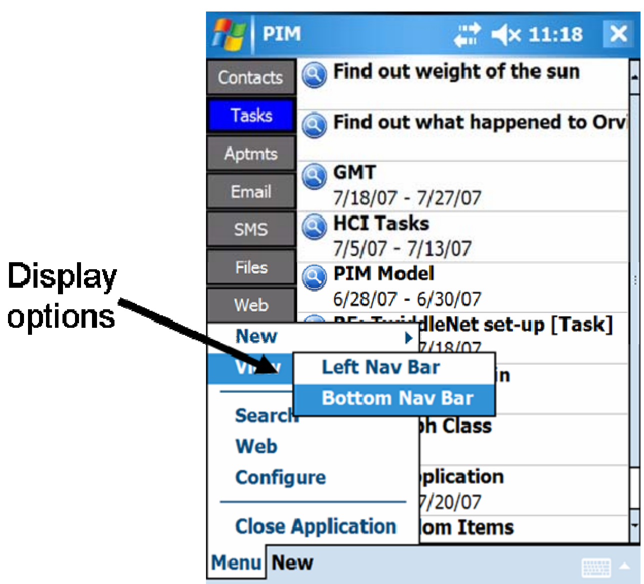

Figure 2. Menu for selecting the desired interface.

The basic menu layout in both cases allows the user to toggle rapidly through the various item types on the device without the hassle of minimizing and maximizing screens repeatedly. However, the navigation tabs on the left side of the screen provide room for additional options that will not fit as icons on the lower navigation bar. The “Web” and “Search” options do not have icons on the lower navigation bar, but they are available as menu items if the user clicks “Menu.” Figure 3 shows how the “Web” and “Search” options can be accessed when the lower navigation bar is used.

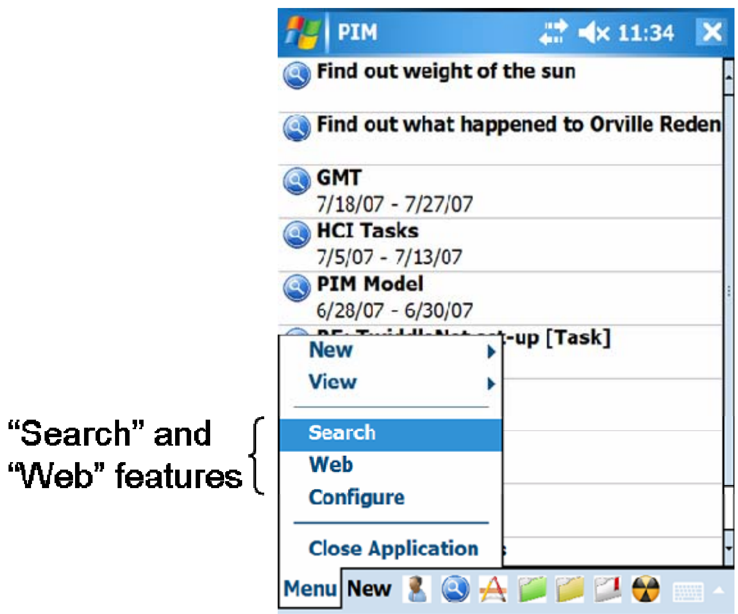

Figure 3. “Web” and “Search” options are still available even though an icon is not placed at the bottom of the screen.

With either view selected, the user can manipulate all data here, rather than in multiple source applications. To ensure no work is ever lost, work done in the original applications is available in MPM, and work done in MPM is available in the original application. For example, if a user creates an appointment in MPM, it is visible in the normal calendar, as well as the “Today” screen if the user has it configured to view appointments. This lets the user decide what application is most appropriate, with changes reflected in both locations. However, most functions needed by the user are implemented in MPM to ease all IM activities.

There are parallels between using tabs on the current generation of web browsers and our system in terms of organizing information and ease of navigation. Tabbed browsing allows the user to view multiple web pages without needing to open multiple instances of the browser application. However, browser tabs are generic and have no inherent link with what they are displaying. We designed the interface of MPM with the similar goal of enabling the viewing of multiple types of information without needing to open multiple applications. The key difference is that we designed our tabs to provide a convenient way of locating related information and linking this information. Items of different types are frequently related in some meaningful way to the user and we want the user to be able to capture such relationships. Figure 4 shows the ease on navigation provided by these tabs.

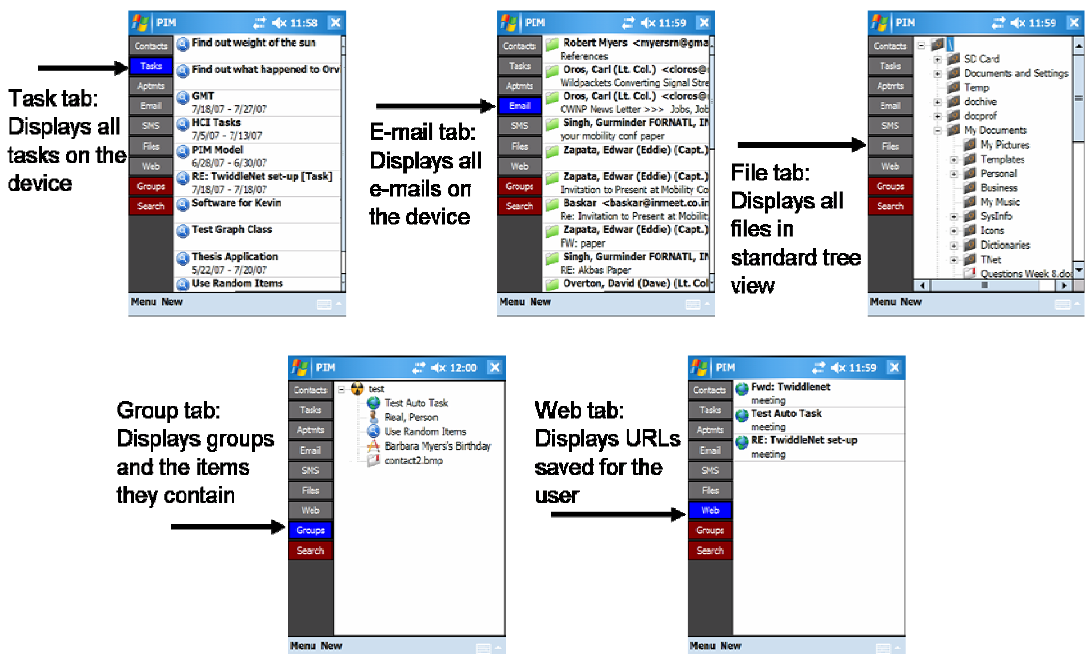

Figure 4. Navigation from Tasks to E-mail to Files to Groups to URLs within the integrated interface.

In addition to improving navigation through several collections of items, MPM also makes viewing the content of individual items more informative to a user. For example, if a user is looking at an e-mail that is logically connected to an appointment and several tasks that are also stored on the device, these connections should be visible to the user when the content of the e-mail is being viewed. Not displaying these links requires the user to recall the related information rather than reminding the user of the connection that the device has already captured in the content of the stored items. Leveraging the fact that relationships among the information exist is useful for displaying more information to the user while he or she is conducting a high-level scan of a single item. We display selected items in a custom form that displays the item content and all items that the user linked to it. We discuss this in more detail in the following section.

D. LINKING AND GROUPING

Expanding on the concept of the task vista [10], allowing a user to group related, yet different typed items is a useful tool for assisting in a rapid, multitask environment. Task lists, e-mail, appointments, supporting files, etc. all play a part in individual objectives such as planning for a conference, or making a sales presentation. However, physical and logical separation of these items is the norm, no matter how well organized a file structure is. This results in the user spending more time searching through his or her scattered information rather than using the information to accomplish his or her true goal, or allowing the user to focus on events taking place in the real world. To alleviate this situation, MPM provides the capability to group items, create logical connections, and visualize those connections. Doing this accomplishes a rough form of "chunking," a technique that allows a person to retain more information in a short period of time [3].

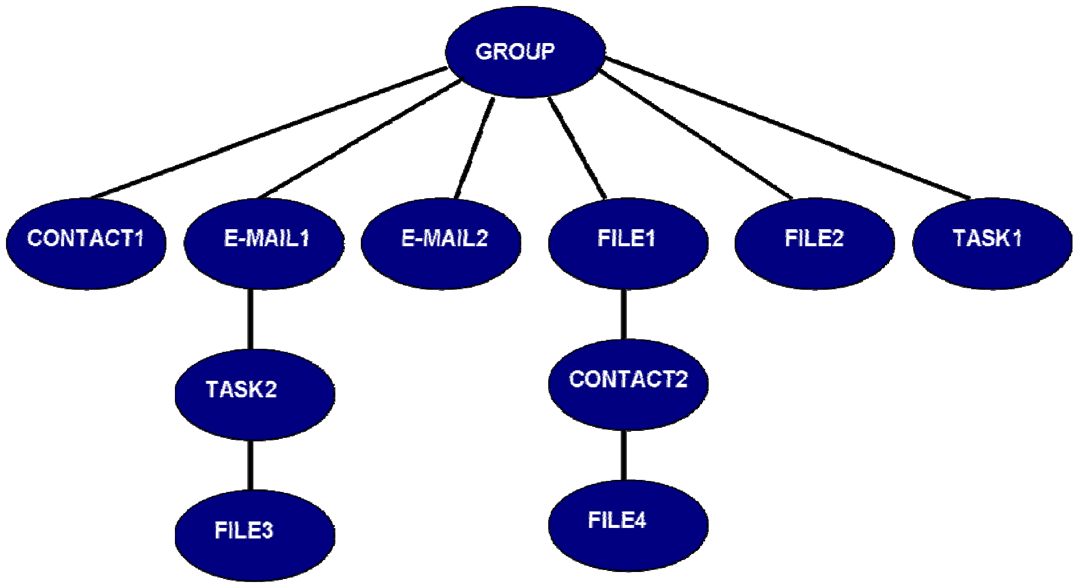

Figure 5. MPM can capture relationships like the ones depicted here. Such links are common among items that are related to tasks and projects. Other handheld IM tools cannot capture such links.

The grouping ability keeps the user from having to develop an elaborate directory structure on the mobile device, and allows the linking of items such as contacts and appointments with items such as files and e-mails. Items such as contacts, appointments, and e-mails are not accessible through the file explorer, so the user cannot place them in a purpose built directory even if he or she was inclined to do so. Figure 5 depicts the possible relationships a user would like to capture, allowing the user in a time constrained mobile scenario to view the key elements of an entire project at a glance while riding an elevator or walking through a lobby. MPM allows the user to create such groups and add items to them. MPM captures these relationships using XML as show in figure 6. Capturing this data allows MPM to render the relationships for the user in the integrated interface as seen in figure 7. Additionally, once MPM captures these relationships, each item is linked to all the other items in the group. These links are visible when any item in the group is displayed.

```xml
<group name="Test Group" type="7">
    <item name="1 PIM Model" type="0" group_name="Test Group" tag="PIM Model">
    </item>
    <item name="1 HCI Tasks" type="0" group_name="Test Group" tag="HCI Tasks">
    </item>
    <item name="feed.xml" type="5" group_name="Test Group" tag="tnet\bin\feed.xml">
    </item>
    <item name="email1.xml" type="5" group_name="Test Group" tag="tnet\bin\email1.xml">
    </item>
    <item name="Real, Person" type="1" group_name="Test Group" tag="Real, Person">
    </item>
    <item name="RE: IS Related Interesting Articles - 7/11/2007" type="3" group_name="Test Group"
                         tag="RE: IS Related Interesting Articles - 7/11/2007">
       <item name="Test Graph Class" type="0" group_name="Test Group" tag="Test Graph Class">
       </item>
       <item name="6.jpg" type="5" group_name="Test Group" tag="My Documents\TNet\shared\6.jpg">
       </item>
    </item>
    <item name="Lost and Found" type="3" group_name="Test Group" tag="Lost and Found">
       <item name="Use Random Items" type="0" group_name="Test Group" tag="Use Random Items">
       </item>
       <item name="VS1.jpg" type="5" group_name="Test Group"
                         tag="My Documents\TNet\shared\VS1.jpg">
       </item>
    </item>
</group>
```

Figure 6. XML representation of the group seen in figure 5.

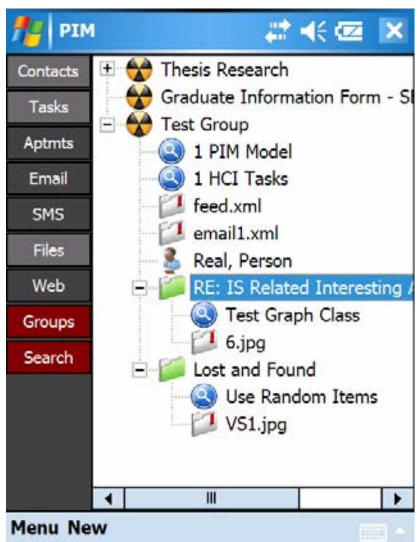

Figure 7. Rendering of the group in the user interface after reading the XML.

E. DISPLAYING LOGICAL LINKS

Figure 8 shows a custom view that visualizes the logical links when the user views the content of an individual item. The content of the item is the primary information displayed, but the user can see all linked items at the bottom of the screen. The user can select the items at the bottom of the screen for viewing or editing without having to transition to another screen. Figure 8 shows the display of a contact with all of the linked items at the bottom of the screen. A user can view all of the items that are in the same group as this contact by simply clicking them, regardless of its type or its location on the device. This facilitates faster retrieval of related items, as well as reminding the user of items that are related, but the user has not viewed recently.

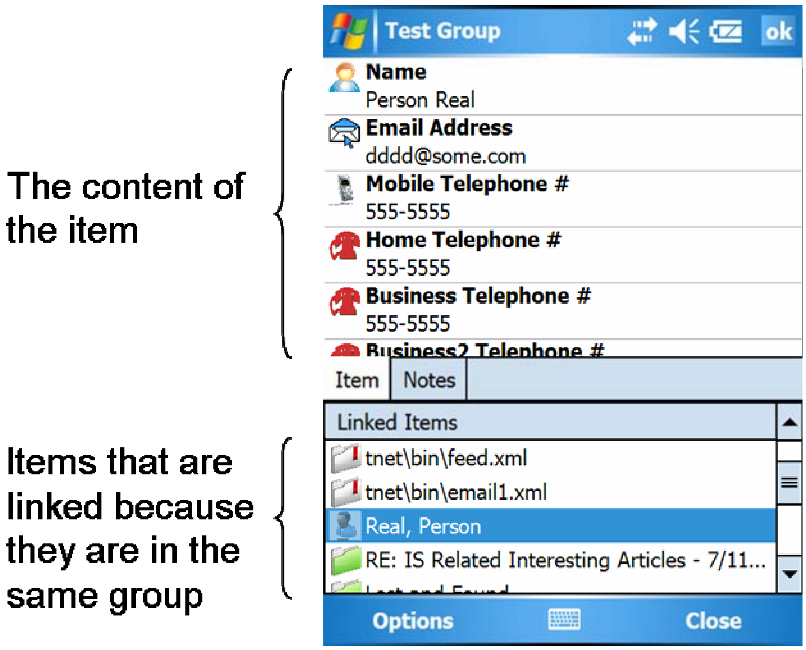

Figure 8. Display of an item’s content with linked items one click away.

Figure 9 shows the ease of navigating among the linked items. MPM displays all items that are associated with the selected item at the bottom of the screen, so they are visible but not the focus of the user’s screen. If the user wants to examine one of the linked items more closely, then he or she can simply click on the item to view its content. Additionally, the user can select any of the linked items for editing while viewing this screen simply by clicking the “Options” menu item as seen in figure 10. Finally, if the item contains a phone number, the user can click on the number to initiate a call if he or she is using a device with a phone capability. Figure 11 shows this feature.

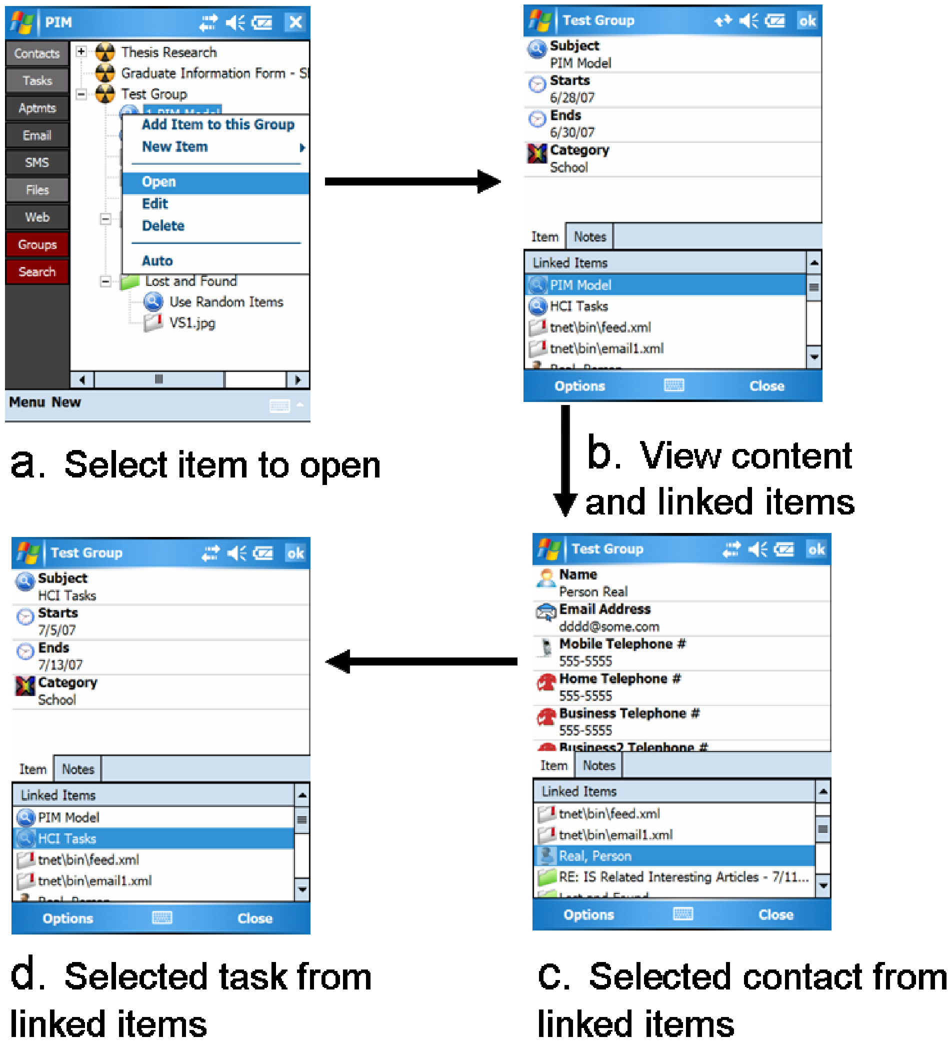

Figure 9. Movement among linked items. a) Item in group opened. b) Content of item and list of linked items. c) Linked contact selected and opened. d) Linked task selected and opened.

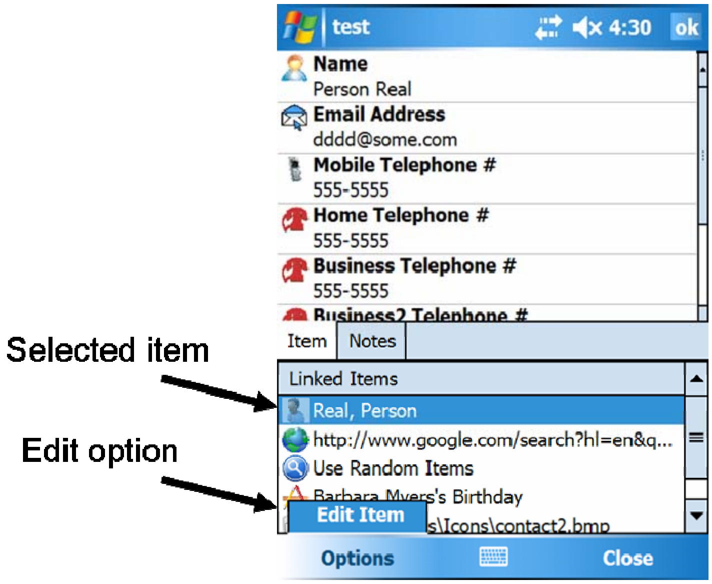

Figure 10. Option to edit the item selected at the bottom of the screen.

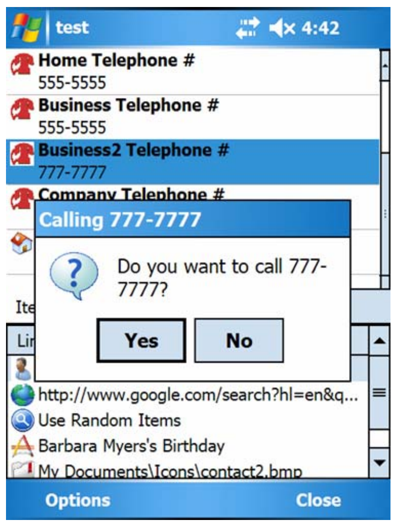

Figure 11. Option to call a number that is contained in an item.

F. AUTOMATIC GROUPING

A person may have many reasons for grouping or linking items. However, there are cases when creating a group automatically is possible. For example, automatic grouping is possible when a primary item implies the creation of other items. We use this primary item as a root for creating a group of new, supporting pieces of information. The scenario we exploit is when the receipt of one item triggers the creation of supporting items. For example, the receipt of an e-mail may warrant the creation of a task, an appointment, or both. The user could manually make the supporting items and type in all of the information needed from the e-mail. However, this process is so tedious on a handheld device, users rarely do it, and the explicit or implied tasks of the e-mail are lost. MPM automates creation and content transfer, so the user does not have to type it in. If the message contains information about a scheduled meeting, the system creates a group for the message and creates a new appointment. Currently, the system simply looks for dates in the message body and triggers the creation of a group and a new appointment, but modifications could be made to add methods that would make the system more robust. We populate the group with the e-mail message and the calendar information for the appointment. A user can view a list of all the items in a group on one screen. If a user selects an item from the group list, the screen will display the content of the item (i.e., the e-mail body) and it will provide a link to the other related item, in this case an appointment.

A similar automation feature allows the user to select an item and choose to make a new related item. For example, if a user wants to make an appointment out of an e-mail, or a task out of an appointment, the user only has to select the root item and choose the new type of item to make. Figure 12 shows this sequence. MPM copies the pertinent information from the root item to the new item, and it creates a group that links the two items. The association between the two items is visible when viewing the existing groups or when the user views the content of one of the items. The goal of this feature is to allow the user to quickly create and populate a new item with minimal typing, while simultaneously capturing the relationship that now exists between the two items. Three stylus clicks will perform the entire process.

The final automation feature of MPM lets a user select a group and auto-populate it with items, using the group's name as search criteria. It is a simple search, but MPM links the results and the user can view them as a group for as long as he or she wants to keep the collection. With two stylus clicks, MPM populates the group with items that contain text strings matching the group's name. Figure 13 shows this process and the search results for the group named “Thesis” on a test device. This is a useful tool for finding related items and saving the linkages without ever typing anything. Unneeded items can be removed with a single stylus click.

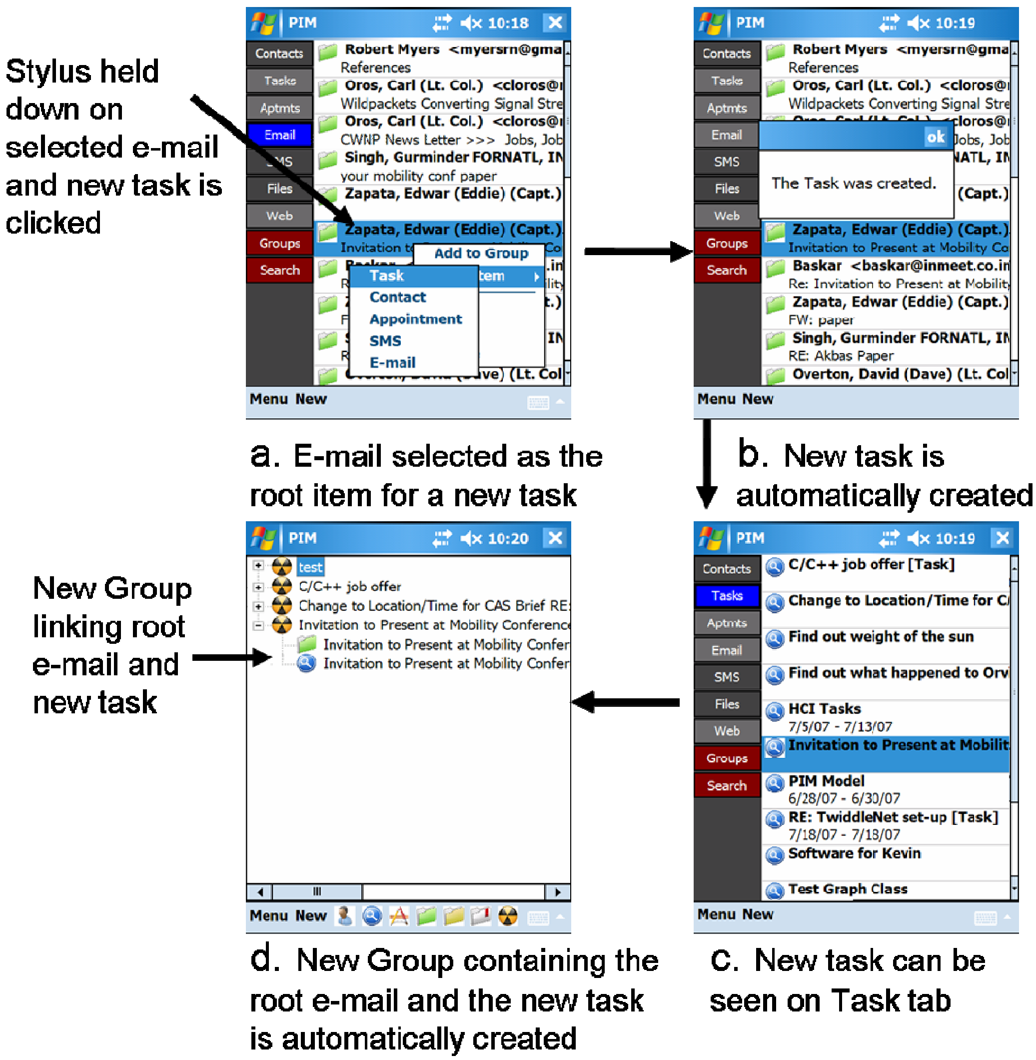

Figure 12. Auto grouping sequence: a) User selects an e-mail as a root item to create a task. b) MPM automatically creates a new task and a new group. c) The task is visible in the task tab. d) The group containing e-mail and auto-generated task

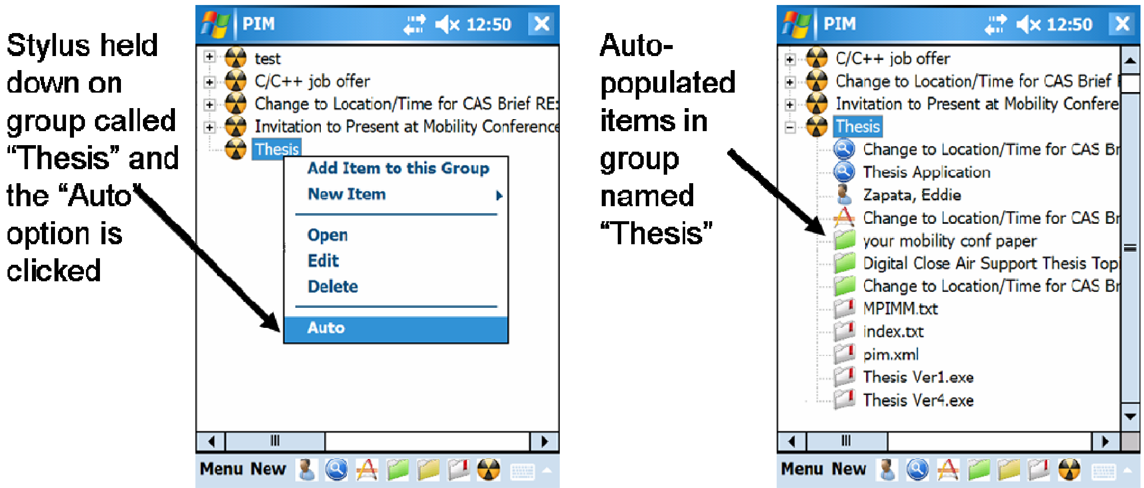

Figure 13. Auto-population of a group named "Thesis."

G. SEARCHING

The final feature included in MPM is an integrated text-based search tool. Even though the user can quickly navigate through the interface, the user may still miss the sought after item. As a result, we built a search tool into MPM. MPM indexes all of the items on the mobile device and writes the metadata to an XML document. MPM does this at startup and every 10 minutes while it is running. Figure 15 shows excerpts of this document. MPM reads in the XML file and searches the tree structure created by the XML, allowing MPM to perform fast queries of all files, e-mails, contacts, etc.

We designed this tool because there is no Google Desktop API or Microsoft Directory Services API equivalent for the Windows Mobile operating system. The XML uses the following schema for indexing the device. The root tag is "index," and all tags are children of this root. "Index" can have the following children: "tasks," "contacts," "appointments," "emails," and "directories." The tree in figure 15 shows the structure of the schema. It provides easy searching because if a user filters on jpegs, the search only has to go down the directories branch.

```
<file type="5" name="Scores" extension="txt"
full_path="\Scores.txt" size="11" last_access_date="3/11/07"
last_access_time="11:00:00 PM">
  <word name="Scores" count="1" />
  <word name="txt" count="1" />
  <word name="25440" count="1" />
  <word name="16" count="1" />
 </file>
- <file type="5" name="MPIMM" extension="txt"
full_path="\MPIMM.txt" size="177" last_access_date="5/7/07"
last_access_time="12:00:00 AM">
  <word name="MPIMM" count="1" />
  <word name="txt" count="1" />
  <word name="Thesis" count="2" />
  <word name="GROUP" count="2" />
  <word name="7" count="3" />
  <word name="CS4923" count="2" />
  <word name="Test" count="4" />
  <word name="Graph" count="4" />
  <word name="Class" count="3" />
  <word name="Class\Miller" count="1" />
  <word name="," count="3" />
  <word name="Kasey" count="3" />
 </file>
```

```
<task type="0" subject="PIM Model" due_date="6/30/07">
  <word name="Going" count="1" />
  <word name="from" count="2" />
  <word name="real" count="1" />
  <word name="world" count="1" />
  <word name="to" count="2" />
  <word name="desktop" count="1" />
  <word name="mobile" count="1" />
  <word name="device." count="1" />
  <word name="Filtering," count="1" />
  <word name="sorting," count="1" />
  <word name="organizing," count="1" />
  <word name="viewing" count="1" />
  <word name="information." count="1" />
  <word name="Grouping:" count="1" />
  <word name="Manual" count="1" />
  <word name="and" count="2" />
  <word name="algorithms" count="1" />
  <word name="connection" count="1" />
  <word name="between" count="1" />
  <word name="items." count="1" />
  <word name="PIM" count="1" />
  <word name="Model" count="1" />
 </task>
```


Figure 14. XML excerpt from the device content index.

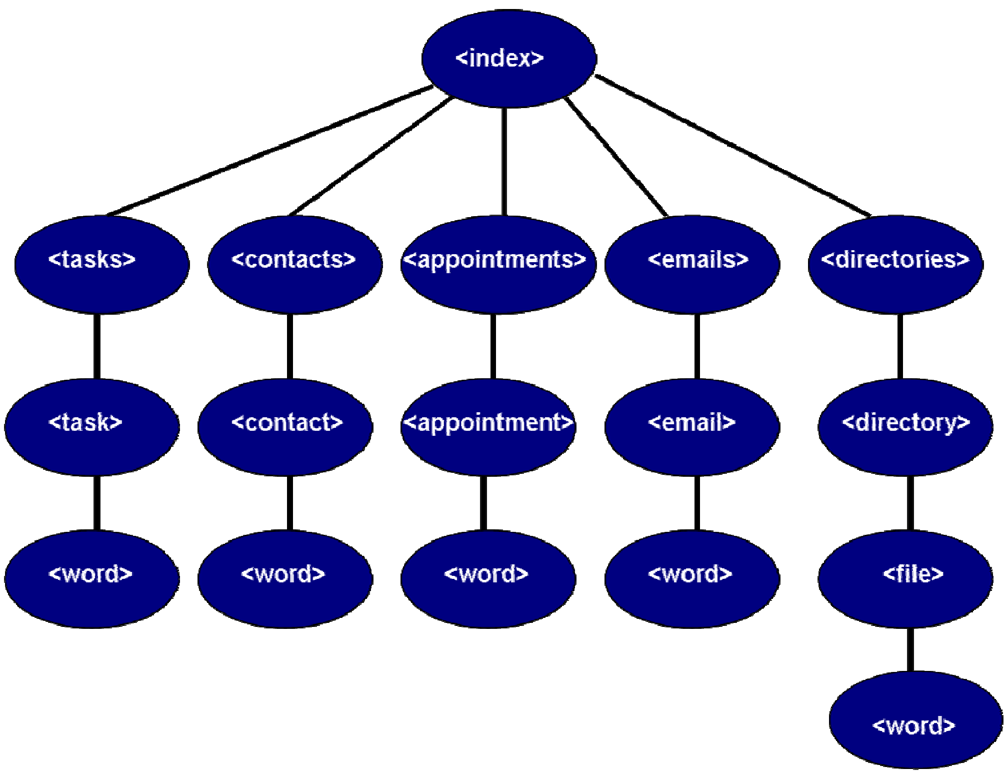

Figure 15. XML tree structure of the device index.

Figure 16 shows the search tool. The user can type in keywords and choose the item type he or she is looking for, such as a contact, jpegs, e-mails, etc. The user can view any item found, and he or she can also take any item in the search results and add it to a group. This enables rapid population of a group from a single screen when the user only has time to type in one or two keywords.

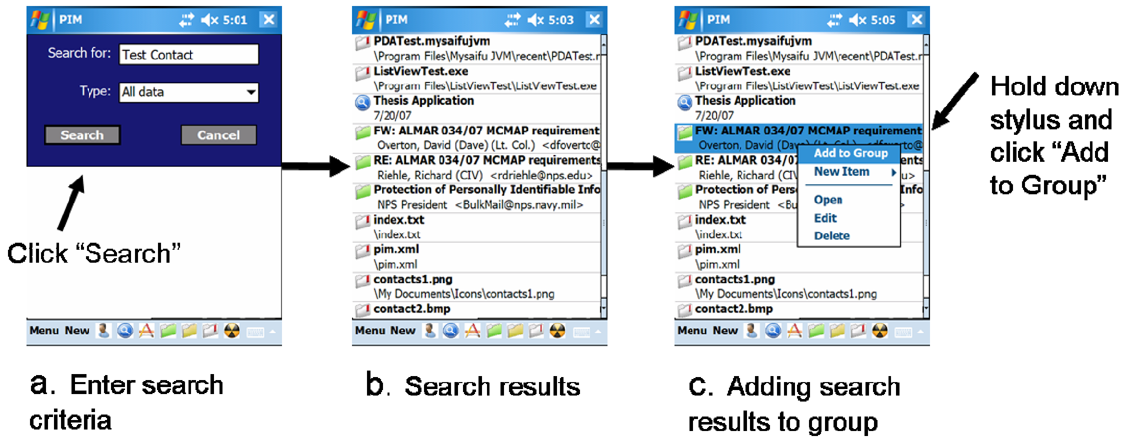

Figure 16. a.) Integrated search tool b.) Search results c.) Selecting an item to add to a group

H. SUMMARY

MPM places several features in a single location. These features include access to all data in a single interface, the ability to group related items and maintain these relationships, the ability to view item content and linked items simultaneously, the ability to generate supporting IM items automatically, and finally an integrated search tool. These features reduce the amount of navigation required on the handheld device, present more meaningful content to the user, and improve information accessibility. The following chapter outlines the process involved in developing the system.

IV. SYSTEM ARCHITECTURE AND IMPLEMENTATION

A. INTRODUCTION

This chapter details the software development of MPM. The previous chapter described what MPM does and how the user interacts with it. This section describes the platform we selected, why we selected it, how we designed the interface, and how we implemented key features. Figure 17 shows the complete class diagram of MPM and we will highlight several classes in later sections. The next section describes how MPM's specifications were determined.

B. SYSTEM SPECIFICATIONS

We determined the specifications of MPM by addressing the following design questions.

- What is the best way to navigate among multiple information stores on a small screen?
- What is the best way to display all of the device's data in a single interface?
- What is the best way store and render relationships among several different items?
- How do we create persistent links between items?
- What IM activities can we automate?
- What information can we create automatically for the user?

The answers to these questions evolved as we examined several interface designs and determined what features to include in MPM. The first two questions deal directly with interface requirements on a small screen. The third and fourth questions deal with the implementation of groups in MPM. We knew we wanted a grouping or piling ability in MPM, so it was the first key feature we included. The last two questions deal with adding automation to MPM to save the user time on several IM tasks. We also decided that the last two questions fundamentally deal with easing organization for the user, so a search feature grew out of these questions as well. The next section discusses the platform we selected for MPM. We follow this with the details of the interface design and the development of the software.

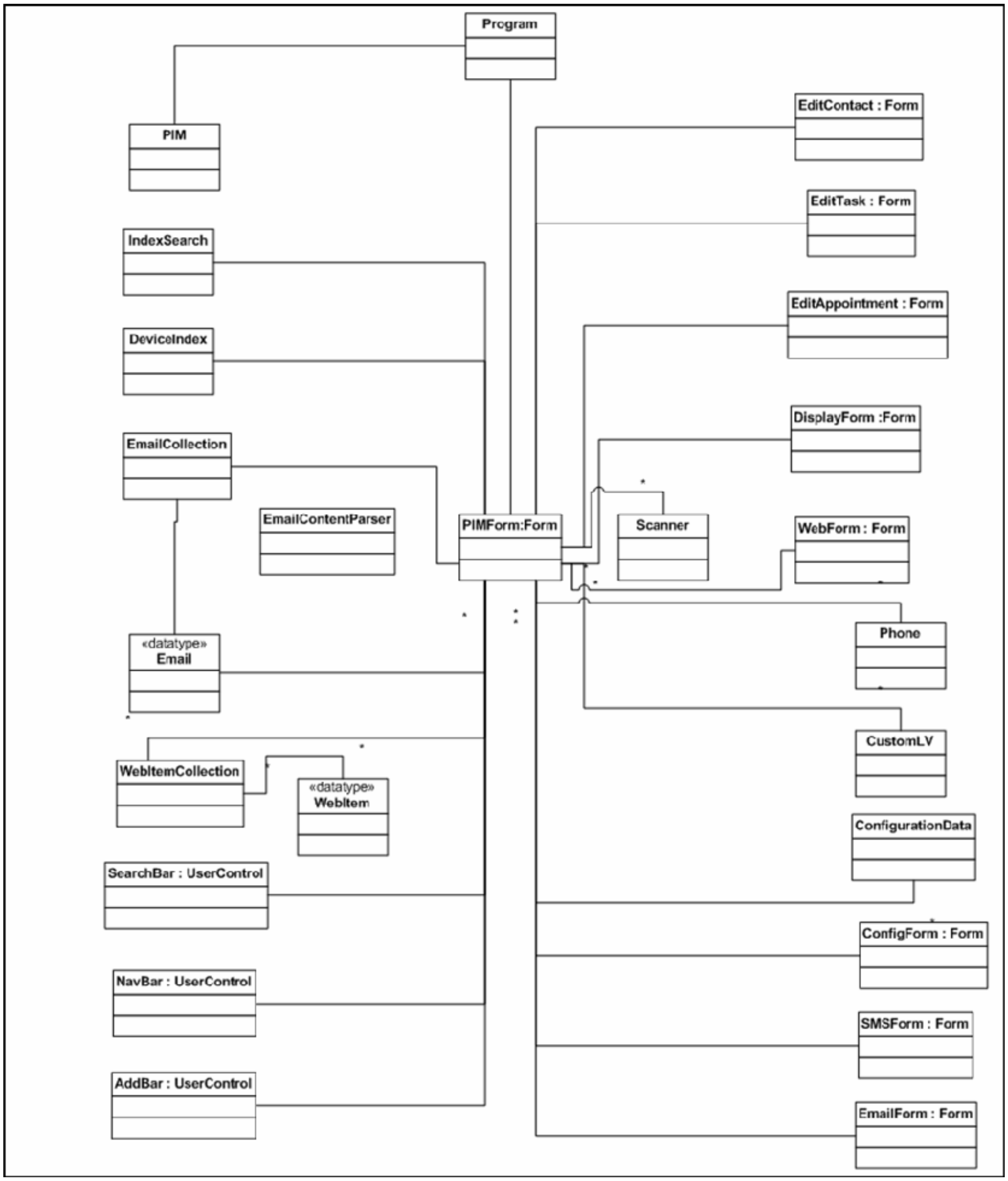

Figure 17. Complete class diagram of MPM

C. PLATFORM SPECIFICATIONS AND CHALLENGES

We developed MPM in C# to run on the Windows .NET Compact Framework (CF). The .Net CF consists of a base of class libraries and a common language runtime (CLR) environment that runs on the Windows Mobile operating systems, starting with Pocket PC 2000 and continuing to the latest version, Windows Mobile 6.0 operating system. We selected this platform because it offers to best balance between hardware independence and providing development tools needed to build MPM. Many hardware platforms do not make their SDKs freely available without purchasing the SDK or being a registered developer for a specific hardware vendor. However, the Windows Mobile SDK is free and includes items such as the Windows PocketOutlook namespace that can access information stores such as contacts, tasks, or appointments. There are no comparable free SDKs or development tools for platforms such as the iPhone, and even if there were that would limit MPM to a single hardware platform. The .Net CF allows us to access this information on any hardware platform running the Windows Mobile operating system. This is not as broad a base of platforms as J2ME might provide, but it is the best mix of capabilities, development tools, and hardware interoperability.

With this selection made, we shifted our focus to addressing the requirements for building a system on a handheld device. The challenges of developing MPM for the Windows Mobile platform are the same as for any other mobile platform. There are several hardware limitations on handheld devices, but these limitations are relatively consistent across many models of handhelds. Screen sizes and resolutions are all comparable, input with a stylus, touch screen, or small keyboard are all about the same, and battery technologies are all relatively standard. The greatest hardware variability is with memory and processor speed. The devices targeted vary from 10 to 40 MB of available memory for the executing program, and the processor speeds vary from 300 to

624 MHz, but even this variability is not that great. Even though MPM is built for the Windows .NET CF, the design will work equally well on any comparable device.

D. USER INTERFACE DESIGN

The design of the user interface went through several stages of development. We started with a main screen that had buttons linking to all of the devices information. Figure 18 displays the original design. However, this provided no simple way to move among the various items. For example, if a user opens the contacts window, a link back to the main screen could be included, but what if the user does not want to go back to the main screen. We had to address this to provide adequate answers to our first two design questions.

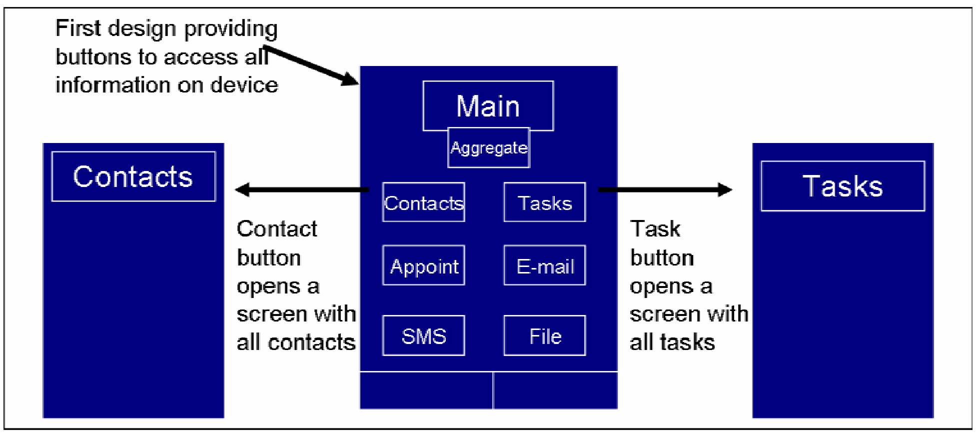

Figure 18. Original design sketch of the main interface.

The answer we came up with was to design a tabbed browsing scheme on an interface that could access all device data. Figure 19 shows the initial sketch. We turned this sketch into a working interface using C# and the classes available in the Compact Framework library. Specifically, we extended the class Form and used it as the primary object of user interaction, and we use the OutlookSession class from the PocketOutlook namespace to access all PIM items on the device.

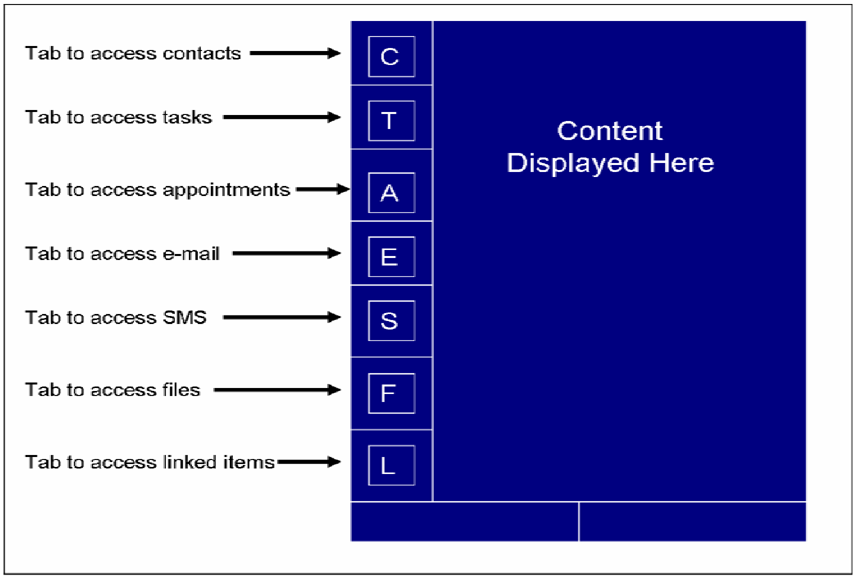

Figure 19. Sketch of tabbed browsing interface.

The first working interface only had limited access to the content of individual items and it used tabs at the bottom of the screen that ran too far to the right. This first version was inadequate, but it slowly evolved as seen in Figure 20. The final version, shown in Figure 21, allows full interaction with all the data and rapid navigation among all item types.

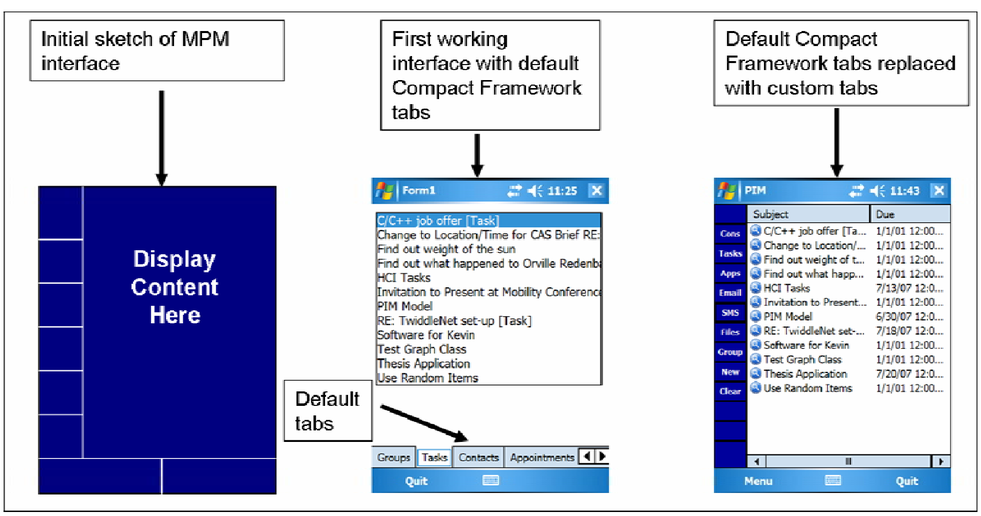

Figure 20. Development from sketch to initial design.

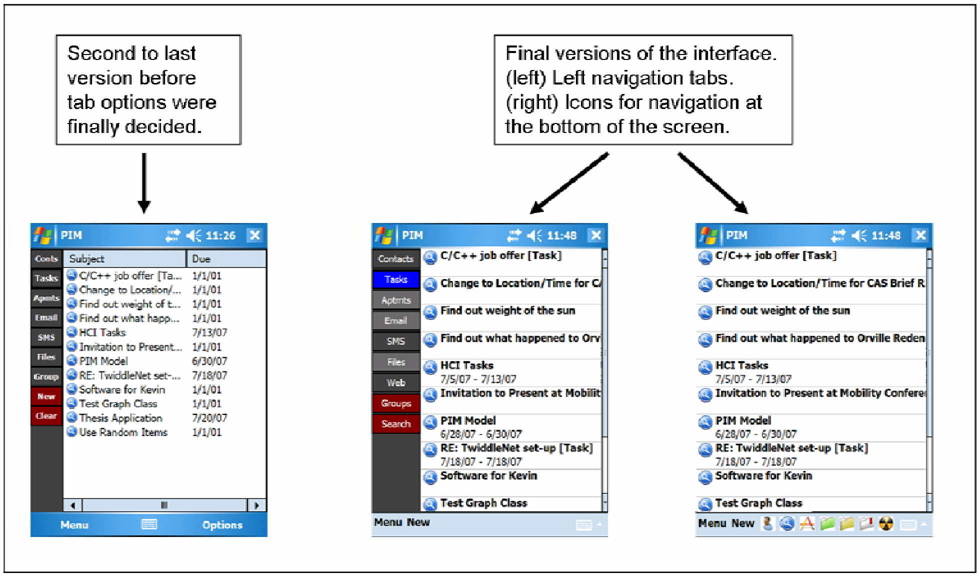

Figure 21. Development from intermediate to final design.

E. IMPLEMENTING GROUPS AND ITEM LINKS

Examination of the group concept showed that it could be implemented using a tree structure. We decided to represent all groups and items of any type as nodes on a tree with all the information needed to distinguish the item saved as properties of the node object. For example, if a user adds a contact to a group, we represent the contact object as a node and add it as a child to the group node. We can link items easily by converting them to node objects. We also use this tree structure for displaying the group items. The .NET CF TreeView and TreeNode classes provide a simple way to represent items as nodes and link them together. One TreeView object exists during the running of MPM, and nodes are added and removed as required.

In addition to the TreeView and TreeNode classes, we use the Dictionary container class to store data about each item in a group. We use the TreeView and TreeNode classes to display a group of linked items, but if the user selects a single item, such as a task, we use the Dictionary to quickly find the items linked to this task. MPM uses this information to display the appropriate links when an item's content is displayed. Figure 22 shows a short code listing for adding a node to the parent node, and it shows what information is placed in the Dictionary for each member of a group.

```
private void AddNodeToTree(String text, String tag, TreeNode parentNode)
    {
        TreeNode newNode        = new TreeNode(text);
        newNode.Tag             = tag;
        parentNode.Nodes.Add(newNode);
    }


internal void AddItemToDictionary(String key, Object item, int number, int flag)
    {
        try
        {
            if (!itemData.ContainsKey(key))
            {
                ArrayList list1 = new ArrayList(5);
                list1.Add(item);
                list1.Add(flag);
                list1.Add("");         // Used for group name.
                list1.Add("");         // Full Path Name for Group: e.g. Meeting/Email5/Task3.
                list1.Add(number);  // Counter, used if item is member of multiple groups.
                itemData.Add(key, list1);
            }
        catch (Exception) { }
    }
```

Figure 22. Code listing of node creation and population of group information in a dictionary.

MPM converts the nodes of the TreeView to XML for storing groups when the application is closed. We selected XML as the format for saving the user's groups for two reasons. First, well-formed XML has the same tree structure as the groups, so it is easy to store the nodes in the tree as XML tags and it is easy to read in the XML and recreate the tree and nodes of the groups when MPM starts. Second, XML is platform independent, so a user can view or search this information on his laptop or desktop. We included this platform independence for possible future extensions, such as making the device easily searchable by systems described by Akbas and Singh [18]. Appendix B shows an example of the XML representation of several groups.

F. IMPLEMENTING AUTOMATIC FEATURES

The last two design questions resulted in MPM having a few automatic features to help the user. We decided to automate the following activities. First, we automated task and appointment creation when an e-mail arrives. Second, we automated the creation of Google queries if the e-mail contains keywords entered by the user. Finally, we automated group population. We implement these features with three key classes: Scanner, DeviceIndex, and IndexSearch.

MPM uses the Scanner class to implement the first feature. The Scanner class uses the FileSystemWatcher class of the OpenNETCF namespace to detect when a new e-mail arrives. Figure 23 shows the setup of this class, so the arrival of a new e-mail triggers the Scanner class. Once this happens, MPM parses the e-mail. The Scanner class creates an appointment if any date or time information is in the text of the e-mail. If date or time information and directive words such as “must,” “due,” “you must,” etc. are in the text, MPM creates a new task. In both cases, MPM creates a new group, placing the e-mail and the newly created item(s) in this group.

Directory containing e-mails

```
/// <summary>
/// <b>Sets the FileSystemWatcher to monitor the "\\Windows\\Messaging" directory.</b>
/// </summary>
private void InitializeFSW()
{
    fileSystemWatcher                           = new FileSystemWatcher(PIM.MESSAGING, "*.*");
    fileSystemWatcher.IncludeSubdirectories = false;
    fileSystemWatcher.NotifyFilter              = NotifyFilters.CreationTime |
                                                   NotifyFilters.DirectoryName |
                                                   NotifyFilters.FileName |
                                                   NotifyFilters.LastWrite |
                                                   NotifyFilters.Size;
    fileSystemWatcher.EnableRaisingEvents   = true;
    fileSystemWatcher.Changed                  += new FileSystemEventHandler(OnCreated);
}
```

Figure 23. Code listing for setting up the FileSystemWatcher.

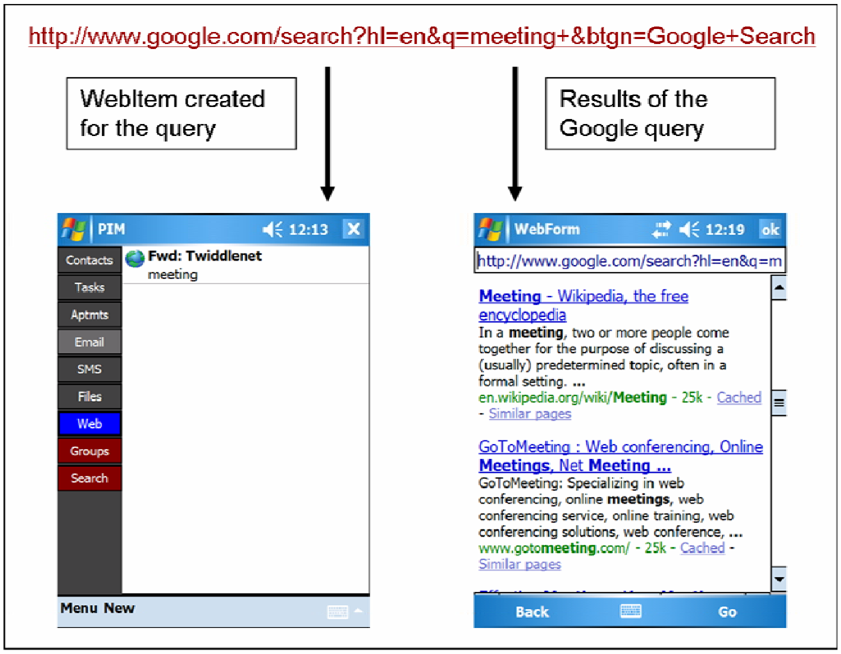

Figure 24. Google query automatically created for a keyword found in an e-mail.

MPM also uses the Scanner class to determine if a new e-mail contains any keywords that the user has entered. If the e-mail contains any keywords, MPM places them in a string to create a Google query. MPM also creates a WebItem, so the user can view the query in the main interface, view the search results, or add it to a group. Figure 24 shows the results of this feature.

MPM uses the DeviceIndex and IndexSearch classes to implement the final automation feature. DeviceIndex scans the entire device and creates an index in XML of all of the device’s data. Appendix C shows a complete example of this index. IndexSearch rapidly searches this XML index. MPM needs this index to rapidly find items because searching the device directly each time an item must be found is too time consuming. Auto-populating a group requires MPM to search every item on the device to find items that contain the groups name. This is essentially a string query, but MPM links the results are together in a group that the user can keep for future reference.

G. IMPLEMENTING THE SEARCH FEATURE

We built the search tool as the final feature to address the last two design questions. As stated in the previous section, the DeviceIndex class creates an XML index of the entire device. MPM searches the XML by creating an XPath query based on the search string entered by the user and any filters entered by the user. Figure 25 shows how the MPM creates an XPath query and Figure 26 shows how MPM uses it to search the designated XML. This XML could also be searched from a desktop or laptop if the system was extended for remote querying.

```
private string CreateXPathExpression(string searchString, string fileType)
{
      string exp = "";
      foreach (string type in itemTypes)
      {
         if (fileType == type)
         {
            return exp = "//" + fileType + "//word";
         }
      }
      if (fileType == "All data" || fileType == "")
      {
         return exp = "//word";
      }
      else
      {
         return exp = "//file[@extension='" + fileType + "']";
      }
}
```

Figure 25. Code listing for XPath query creation.

```
private void ScanXMLNodes(string XPathexpression, string searchString)
{
      string lastNode      = "";
      foreach (XmlNode xNode in xDoc.DocumentElement.SelectNodes(XPathexpression))
      {
         int count     = 0;
         if ((count = PIM.FindAny(searchString, xNode.Attributes[0].Value, 0)) > -1)
         {
            if (lastNode != xNode.ParentNode.Attributes[1].Value)
            {
               lastNode      = xNode.ParentNode.Attributes[1].Value;
               TreeNode tn = new TreeNode(xNode.ParentNode.Attributes[1].Value);
            }
         }
      }
}
```

Figure 26. Code listing for searching XML using an XPath query.

H. SUMMARY

Six design questions drove the design and software development of MPM. The Windows .NET CF platform best supported our goals for MPM and is representative of a large family of handheld devices. The design of the interface resulted from the gradual customization of the default tools available to a .NET CF developer. Finally, the tree structure inherent in grouping resulted in the use of data structures that support that structure. Specifically, the use of TreeViews, TreeNodes, and XML figured prominently in the representation and storage of groups. Additionally, XML served as an excellent tool for indexing the information on a device and using that index for rapid searching. After we finished designing the system, we conducted a usability survey. The results of the evaluation are presented in the next chapter.

THIS PAGE INTENTIONALLY LEFT BLANK

V. SYSTEM EVALUATION

A. INTRODUCTION

We have conducted a limited qualitative user evaluation of our system. Ten participants evaluated and compared the system to the default applications that manage tasks, appointments, contacts, e-mail, searching files, and viewing files. The participants were all graduate level students in the Computer Science curriculum, and most have had minimal interaction with Smartphones or PocketPCs.

We began the evaluation by providing each participant with verbal and hands on instruction on the use of both MPM and the device's native applications. We then asked them to perform a series of ten tasks on the native applications, then in MPM, and rate the processes based on the survey questions that we provided. Appendix A shows the survey form used to conduct the evaluation.

The following sections describe the evaluation process in detail. We include user comments and suggestions for improvement along with two rating tables that show how users rated MPM in comparison to the native applications and what they thought about MPM's features.

B. LEARNING THE SYSTEM

Part of our system evaluation took into account how easy or difficult it was to learn to use MPM. Eight out of ten participants had no experience using a PocketPC device and the other two had limited experience. The eight participants that had no previous interaction with the device were given a period of instruction on how to use the native PIM tools and how to use MPM. The other two members only needed instruction on how to use MPM. The group of eight stated that MPM's unified interface made it easier to learn than the native PIM tools. The main conflict that arose was that once users were taught how to navigate through the native PIM tools, they got used to clicking on the “X” to navigate between applications so users sometimes closed MPM accidentally, forgetting that they could just tab through the various tools. The two participants that had previously used a PocketPC device found MPM easy to learn, but they also found themselves exiting the system by accident. In order to address the issue, we will be adding a prompt screen that will verify if the user truly wants to exit the system. Some of the users also commented that an instruction booklet and more time to familiarize themselves with MPM would have been more beneficial.

C. INFORMATION READABILITY

An important design concern when developing a system for a handheld device is the small amount of screen real estate. As the participants navigated through the various screens, we asked questions regarding the readability of the information. All of the participants said that the information was displayed in a format that could be read quickly and easily. The text was of adequate size and the information displayed on the screen was sufficiently informative but not overwhelming. Most participants also stated that the consistency of the interface allowed them to focus on important information and evaluate it rapidly. The majority of the users preferred the tab bar configuration to the icon configuration because they said that they needed more time with the system in order to become more familiar with what each icon represented. A few felt that a number of labels on the sub-menus lacked clarity. Another issue that was noticed by two users was information running off the screen. The readability issues that arose were minor and will be fixed in the follow-on prototype.

D. RETRIEVAL OF INFORMATION

Another element that we evaluated was whether or not MPM allowed users to retrieve information effectively. All of the participants gave positive feedback about the retrieval function of the system, specifically the system's information linking capability. They liked that they were able to visualize relationships between related information regardless of the document type. One of the participants stated that he liked not having to remember whether there was other information related to his original query and if there was, where did he have to look to find it. The linking capability and visualization of the links made the retrieval process easier for every user. The only complaint was that it took work on the user's part to create the relationships when the information was being stored.

E. SYSTEM NAVIGATION

All of the participants found MPM easy to navigate through. One user said that the tab bar and unified interface made the process of shuffling through various PIM items less complicated. Another commented that after learning to navigate through both the native applications and MPM, it was much faster and easier to navigate through the various PIM tools using MPM. Several participants commented on the ease with which they were able to jump between related information types regardless of whether they were PIM items or file documents. Two navigation issues that were observed were already identified as issues in other sections as well. The first one was confusion based on the labeling of the sub-menu items. The second issue was accidentally exiting the system due to users being accustomed to clicking on the "X" at the top right corner of the screen in order minimize the screen to access other applications.

F. SEARCH CAPABILITY

Most participants stated that they liked the view that the group display provided because of how much it facilitated searching. One user said that the relationships that the group screen presented helped him to visually search and locate items quickly. The text-based search option that MPM provides also received a positive response from the participants. Even though Windows Mobile

5.0 has a search feature, it is not readily accessible, and users were not able to find documents according to their file extensions. All of the participants said that they liked that MPM had a search tab and they enjoyed being able to locate items based on the file extension.

G. EFFECTIVENESS OF MANUAL GROUPING

The feature that received the most positive feedback was MPM's grouping capability. All of the participants found the ability to manually group items and view the created relationships extremely useful. None of the users reported ever having seen or heard of a tool that provided a similar capability. Several users commented on how much they liked being able to link PIM items with file documents, and a number of participants also said that they liked having the ability to create PIM items from root items and, the ability to populate groups any necessary document.

H. OVERALL SATISFACTION

All participants found the unique features that our system provided to be extremely useful. A comparison rating was done on a seven point scale ranging from 1 – Easier, 4 – Same, 7 – Harder. The tasks assigned and the feedback received is shown in Table 1.

The usefulness of MPM's unique features was also rated by the participants on a seven point scale: 1 – Useful, 7 Not Useful. The results are shown in Table 2.

The results demonstrate that in the case in which the user needs to create new individual items, our system was marginally better than the native applications. As groups became populated, the ease of navigating among linked items became quite apparent to the participants. Having the search feature built into the application also prompted positive feedback. There were no native application functions to compare to MPM's grouping capability, so informal feedback was received from the participants. This feedback indicated that long-term analysis of the automation is needed before any claims can be made.

Task	**Average Response**
Create a new task called “Task 1” and a new task called “Task 2.”	3
Create an appointment related to “Task 2” called “Appointment 1.”	3
Create a contact called “Contact 1.”	4
Identify in “Task 2” that it involves “Contact 1.”	3
Set “Task 2” to be one day before “Appointment 1.”	2
Delete “Task 1 .”	4
Edit “Task 2”: change the category.	3
View “Task 2,” “Contact 1,” and “Appointment 1.”	1
Find a JPEG image.	1
Identify in “Task 2” that it is related to this image.	1

Table 1. Tasks performed by participants in native applications and in MPM. Responses made on a seven point scale: 1 – Easier in MPM, 4 – Same, 7 – Harder in MPM.

MPM Feature	Average Response
Ability to view groups.	1
Ability to see relationships when item content is viewed.	2
Ability to change navigation view: left vs. lower tabs.	1
Ability to conduct text-based searches	3
Ability to create a new PIM item from a source document or item	2
Usefulness of the unified interface	1

Table 2. MPM features rated by participants. Responses made on a seven point scale: 1 – Useful, 7 – Not Useful.

VI. SUMMARY AND CONCLUSIONS

A. SUMMARY

We accomplished two main goals during our thesis research process. First, based on the research, interviews and personal experience, we identified some of the challenges related to information management, and analyzed how they correlate to the mobile environment. Then we designed and presented a system that makes managing information on a mobile device simpler and potentially more attractive to the mobile user. This section recapitulates the findings that were ascertained during our investigation of the information management topic, and reiterates the design concepts that we implemented to address some of the issues that were identified.

We developed a system, MPM, that runs on the Microsoft Windows Mobile 5.0 PocketPC Phone Edition platform. MPM presents a single interface for navigating through the various information banks of the mobile device without overwhelming the user or the device. MPM also implements a logical linking capability that allows users to create groups containing any kind of information regardless of the PIM or document type. The established relationships can then be viewed by selecting group tab on the system's navigation bar. We also implemented a number of automatic features for simplifying item creation and grouping. Finally, we built a text-based search tool into the system that enables users to search the device without having to start a new application. The user can then add the documents acquired from the query to any group that he or she creates. The features implemented in MPM increase the ease of navigation required on a handheld device, present meaningful content to the user, and improve information accessibility.

The following section presents our recommendations for future work based on the data that was gathered during the usability testing of the system, and analysis of current mobile PIM research.

B. FUTURE WORK

We present four possible areas of improvement to the existing properties and features of the system: (1) interface, (2) email access, (3) group creation, (4) automated web queries. We then present three recommendations for enhancement of the current system: (1) context awareness, (2) distributed searching, (3) additional platforms.

1. Improvements to Current Features

In response to user feedback, a few modifications need to be made to the current interface. The first is to improve the custom list view. In some views, users noted that information ran off the screen, and there were also times when the bottom portion of the list view did not extend to cover the entire screen. The issue of information running off the screen can be corrected by adding a horizontal scroll bar to the list view. Additionally, the list view paint method needs to be modified so the list view covers the appropriate view dimensions. Improvement also needs to be made to the stylus click response so that system reacts quicker to selected options. Finally, the option to place the navigation bar on the right side of the screen needs to be provided to facilitate single hand navigation.

MPM currently displays every e-mail in the windows messaging directory, but it does not separate them according to the source account (i.e., Gmail, hotmail, outlook, etc.) or by folder of origin (i.e., Sent, received, deleted). We need to make modifications to the system so it can provide the user with those distinctions. Also, the ability to sort e-mails according to date, from, size, etc., needs to be added.

At present, natural language processing is not being done by the system. As a result, even though appointments and tasks are being created based on information contained in received e-mails, the dates and reminders are sometimes inaccurate. Natural language processing could also assist in optimizing our automated web query idea. We propose the idea of initiating automatic web searches based on information extracted from an email or an SMS. This capability would eliminate the need for the user to deal with menial tasks. One scenario deals with receiving an e-mail about a conference in a distant city. With natural language processing, the information about the location and dates could be extracted. Based on that information, the system could provide information from the internet about weather conditions for the specified dates, information about flights to the destination along with hotel information.

Finally, we developed our system to run specifically on the Windows Mobile 5.0 PocketPC platform, but the system can be modified to run on other smartphones.

2. Features to Add

The main feature that we feel would enhance the capabilities of the system would be the addition of context awareness. This topic is explored by [16] and [17]; each presents a different perspective on how the concept could be implemented. Both implementations would add greatly to our system. Another addition that would improve on our current system is the ability to conduct PIM queries across multiple devices as discussed by [18]. Distributed searching would enable linking regardless of PIM type or storage location.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A: USABILITY SURVEY

List of tasks to perform on device

Tasks using native applications

1. Create a new task called “Task 1” and a new task called “Task 2.”

2. Create an appointment related to “Task 2” called “Appointment 1.”

3. Create a new contact called “Contact 1.”

4. Identify in “Task 2” that it involves “Contact 1.”

5. Delete “Task 1.”

6. Edit “Task 2”: change the category.

7. Set the “Task 2” to be one day before “Appointment 1.”

8. View Task 2, Contact 1, and Appointment 1.

9. Find any JPEG image.

10. Identify in “Task 2” that it is related to the JPEG image.

Tasks using MPM

1. Create a new task called “Task 1” and a new task called “Task 2.”

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

2. Create an appointment related to “Task 2” called “Appointment 1.”

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

3. Create a new contact called “Contact 1.”

1----------2----------3----------4----------5----------6----------7

Easier Same Harder

4. Identify in “Task 2” that it involves “Contact 1.”

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

5. Set the “Task 2” to be one day before “Appointment 1.”

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

6. Delete “Task 1.”

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

7. Edit “Task 2”: change the category.

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

8. View Task 2, Contact 1, and Appointment 1.

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

9. Find any JPEG image.

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

10. Identify in “Task 2” that it is related to the JPEG image.

1----------2----------3----------4----------5----------6----------7
Easier Same Harder

MPM Specific Tasks

1. Ability to view groups.

1----------2----------3----------4----------5----------6----------7
Useful Not Useful

2. Ability to see relationships when item content is viewed.

1----------2----------3----------4----------5----------6----------7
Useful Not Useful

3. Ability to change navigation view: left vs. lower tabs.

1----------2----------3----------4----------5----------6----------7
Useful Not Useful

4. Ability to conduct text-based searches.

1----------2----------3----------4----------5----------6----------7
Useful Not Useful

5. Ability to create a new PIM item from a source document or item.

1----------2----------3----------4----------5----------6----------7
Useful Not Useful

6. Usefulness of the unified interface.

1----------2----------3----------4----------5----------6----------7
Useful Not Useful

7. How do you compare ease of navigation between the two methods?

8. Were there any areas of confusion or features that were not intuitive?

9. Do you have any suggestions for improvement?

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B: GROUP XML


```xml
<?xml version="1.0" encoding="utf-8" ?>
<groups>
<group name="Thesis Research" type="7">
<name>Thesis Research</name>
<item name="PIM Model" type="0" group_name="Thesis Research" tag="PIM Model">
<name>PIM Model</name>
</item>
<item name="Thesis Application" type="0" group_name="Thesis Research" tag="Thesis Application">
<name>Thesis Application</name>
</item>
<item name="Zapata, Eddie" type="1" group_name="Thesis Research" tag="Zapata, Eddie">
<name>Zapata, Eddie</name>
</item>
<item name="HCI Tasks" type="0" group_name="Thesis Research" tag="HCI Tasks">
<name>HCI Tasks</name>
</item>
</group>
<group name="Graduate Information Form - SEPT 2007" type="7">
<name>Graduate Information Form - SEPT 2007</name>
</group>
<group name="Test Group" type="7">
<name>Test Group</name>
<item name="1 PIM Model" type="0" group_name="Test Group" tag="PIM Model">
<name>1 PIM Model</name>
</item>
<item name="1 HCI Tasks" type="0" group_name="Test Group" tag="HCI Tasks">
<name>1 HCI Tasks</name>
</item>
<item name="feed.xml" type="5" group_name="Test Group" tag="tnet\bin\feed.xml">
<name>feed.xml</name>
</item>
<item name="email1.xml" type="5" group_name="Test Group" tag="tnet\bin\email1.xml">
<name>email1.xml</name>
</item>
<item name="Real, Person" type="1" group_name="Test Group" tag="Real, Person">
<name>Real, Person</name>
</item>
<item name="RE: IS Related Interesting Articles - 7/11/2007" type="3" group_name="Test Group" tag="RE: IS Related Interesting Articles - 7/11/2007">
<name>RE: IS Related Interesting Articles - 7/11/2007</name>
```

```
<item name="Test Graph Class" type="0" group_name="Test Group" tag="Test Graph Class">
<name>Test Graph Class</name>
  </item>
<item name="6.jpg" type="5" group_name="Test Group" tag="My Documents\TNet\shared\6.jpg">
<name>6.jpg</name>
  </item>
  </item>
<item name="Lost and Found" type="3" group_name="Test Group" tag="Lost and Found">
<name>Lost and Found</name>
<item name="Use Random Items" type="0" group_name="Test Group" tag="Use Random Items">
<name>Use Random Items</name>
  </item>
<item name="VS1.jpg" type="5" group_name="Test Group" tag="My Documents\TNet\shared\VS1.jpg">
<name>VS1.jpg</name>
  </item>
  </item>
  </group>
  </groups>
```

APPENDIX C: INDEX XML


```xml
<?xml version="1.0" encoding="utf-8" ?>
<index>
<tasks>
<task type="0" subject="" due_date="1/1/01">
<word name="" count="2" />
  </task>
  </tasks>
<contacts />
<appointments>
<appointment type="2" subject="appointment1" date="8/8/07" location="">
<word name="" count="1" />
<word name="appointment1" count="1" />
  </appointment>
<appointment type="2" subject="bob" date="8/8/07" location="">
<word name="" count="1" />
<word name="bob" count="1" />
  </appointment>
  </appointments>
<emails />
<directories>
<directory name="\">
<file type="5" name="mxip_system" extension="vol" full_path="\mxip_system.vol"
    size="143360" last_access_date="5/21/06" last_access_time="12:00:00 AM">
<word name="mxip_system" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="mxip_notify" extension="vol" full_path="\mxip_notify.vol"
    size="135168" last_access_date="5/21/06" last_access_time="12:00:00 AM">
<word name="mxip_notify" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="mxip_lang" extension="vol" full_path="\mxip_lang.vol"
    size="28672" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="mxip_lang" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="mxip_initdb" extension="vol" full_path="\mxip_initdb.vol"
    size="32768" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="mxip_initdb" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="cemail" extension="vol" full_path="\cemail.vol"
    size="225280" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="cemail" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="pim" extension="vol" full_path="\pim.vol" size="323584"
    last_access_date="8/21/06" last_access_time="12:00:00 AM">
```

```
<word name="pim" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="Microsoft .NET CF 2.0.LOG" extension="TXT"
    full_path="\Microsoft .NET CF 2.0.LOG.TXT" size="7310"
    last_access_date="7/18/07" last_access_time="12:00:00 AM">
<word name="Microsoft .NET CF 2.0.LOG" count="1" />
<word name="TXT" count="1" />
  </file>
<file type="5" name="index" extension="xml" full_path="\index.xml" size="0"
    last_access_date="7/18/07" last_access_time="12:00:00 AM">
<word name="index" count="1" />
<word name="xml" count="1" />
  </file>
<file type="5" name="web" extension="xml" full_path="\web.xml" size="55"
    last_access_date="7/18/07" last_access_time="12:00:00 AM">
<word name="web" count="1" />
<word name="xml" count="1" />
<word name="<?xml" count="1" />
<word name="version="1.0"" count="1" />
<word name="encoding="utf-8"?>" count="1" />
<word name="<webitems" count="1" />
<word name="/>" count="1" />
  </file>
<file type="5" name="pim" extension="xml" full_path="\pim.xml" size="572"
    last_access_date="7/18/07" last_access_time="12:00:00 AM">
<word name="pim" count="1" />
<word name="xml" count="1" />
<word name="<?xml" count="1" />
<word name="version="1.0"" count="1" />
<word name="encoding="utf-8"?>" count="1" />
<word name="<groups>" count="1" />
<word name="" count="100" />
<word name="<group" count="1" />
<word name="name="task2"" count="2" />
<word name="type="7">" count="1" />
<word name="<name>task2</name>" count="2" />
<word name="<item" count="3" />
<word name="type="0"" count="1" />
<word name="group_name="task2"" count="3" />
<word name="tag="task2">" count="1" />
<word name="</item>" count="3" />
<word name="name="task2" count="1" />
<word name="[Appoint]"" count="1" />
<word name="type="2"" count="1" />
<word name="tag="task2" count="1" />
<word name="[Appoint]">" count="1" />
<word name="<name>task2" count="1" />
<word name="[Appoint]</name>" count="1" />
<word name="name="Waterfall.jpg"" count="1" />
<word name="type="5"" count="1" />
<word name="tag="My" count="1" />
```

```
<word name="Documents\My" count="1" />
<word name="Pictures\Waterfall.jpg">" count="1" />
<word name="<name>Waterfall.jpg</name>" count="1" />
<word name="</group>" count="1" />
<word name="</groups>" count="1" />
  </file>
<file type="5" name="MobilePIMMaster" extension="CAB"
    full_path="\MobilePIMMaster.CAB" size="2540607" last_access_date="8/6/07"
    last_access_time="12:00:00 AM">
<word name="MobilePIMMaster" count="1" />
<word name="CAB" count="1" />
  </file>
<file type="5" name="config" extension="xml" full_path="\config.xml" size="0"
    last_access_date="7/24/07" last_access_time="12:00:00 AM">
<word name="config" count="1" />
<word name="xml" count="1" />
  </file>
  </directory>
<directory name="\Documents and Settings">
<file type="5" name="default" extension="vol" full_path="\Documents and
    Settings\default.vol" size="28672" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="default" count="1" />
<word name="vol" count="1" />
  </file>
<file type="5" name="system" extension="hv" full_path="\Documents and
    Settings\system.hv" size="344064" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="system" count="1" />
<word name="hv" count="1" />
  </file>
  </directory>
<directory name="\Documents and Settings\default">
<file type="5" name="user" extension="hv" full_path="\Documents and
    Settings\default\user.hv" size="344064" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="user" count="1" />
<word name="hv" count="1" />
  </file>
  </directory>
<directory name="\Temp" />
<directory name="\My Documents" />
<directory name="\My Documents\My Pictures">
<file type="5" name="Waterfall" extension="jpg" full_path="\My Documents\My
    Pictures\Waterfall.jpg" size="30124" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="Waterfall" count="1" />
<word name="jpg" count="1" />
  </file>
<file type="5" name="Flower" extension="jpg" full_path="\My Documents\My
    Pictures\Flower.jpg" size="26364" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
```

```
<word name="Flower" count="1" />
<word name="jpg" count="1" />
  </file>
<file type="5" name="Birthday_gift" extension="gif" full_path="\My Documents\My Pictures\Birthday_gift.gif" size="39939" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Birthday_gift" count="1" />
<word name="gif" count="1" />
  </file>
<file type="5" name="Heart" extension="gif" full_path="\My Documents\My Pictures\Heart.gif" size="36733" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Heart" count="1" />
<word name="gif" count="1" />
  </file>
<file type="5" name="Photo" extension="gif" full_path="\My Documents\My Pictures\Photo.gif" size="36909" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Photo" count="1" />
<word name="gif" count="1" />
  </file>
<file type="5" name="Pond" extension="jpg" full_path="\My Documents\My Pictures\Pond.jpg" size="26625" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Pond" count="1" />
<word name="jpg" count="1" />
  </file>
<file type="5" name="Utah" extension="png" full_path="\My Documents\My Pictures\Utah.png" size="8681" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Utah" count="1" />
<word name="png" count="1" />
  </file>
  </directory>
<directory name="\My Documents\Templates">
<file type="5" name="Blank Note" extension="pwi" full_path="\My Documents\Templates\Blank Note.pwi" size="0" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Blank Note" count="1" />
<word name="pwi" count="1" />
  </file>
<file type="5" name="Meeting Notes" extension="pwi" full_path="\My Documents\Templates\Meeting Notes.pwi" size="1592" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Meeting Notes" count="1" />
<word name="pwi" count="1" />
  </file>
<file type="5" name="Memo" extension="pwi" full_path="\My Documents\Templates\Memo.pwi" size="2112" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Memo" count="1" />
<word name="pwi" count="1" />
```

```
</file>
<file type="5" name="Phone Memo" extension="pwi" full_path="\My Documents\Templates\Phone Memo.pwi" size="2008" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Phone Memo" count="1" />
<word name="pwi" count="1" />
</file>
<file type="5" name="To Do" extension="pwi" full_path="\My Documents\Templates\To Do.pwi" size="3096" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="To Do" count="1" />
<word name="pwi" count="1" />
</file>
<file type="5" name="Blank Document" extension="dot" full_path="\My Documents\Templates\Blank Document.dot" size="7168" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Blank Document" count="1" />
<word name="dot" count="1" />
</file>
<file type="5" name="Meeting Notes" extension="dot" full_path="\My Documents\Templates\Meeting Notes.dot" size="7168" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Meeting Notes" count="1" />
<word name="dot" count="1" />
</file>
<file type="5" name="Memo" extension="dot" full_path="\My Documents\Templates\Memo.dot" size="7168" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Memo" count="1" />
<word name="dot" count="1" />
</file>
<file type="5" name="Phone Memo" extension="dot" full_path="\My Documents\Templates\Phone Memo.dot" size="7168" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Phone Memo" count="1" />
<word name="dot" count="1" />
</file>
<file type="5" name="To Do" extension="dot" full_path="\My Documents\Templates\To Do.dot" size="14336" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="To Do" count="1" />
<word name="dot" count="1" />
</file>
<file type="5" name="Vehicle Mileage Log" extension="xlt" full_path="\My Documents\Templates\Vehicle Mileage Log.xlt" size="27246" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Vehicle Mileage Log" count="1" />
<word name="xlt" count="1" />
</file>
</directory>
<directory name="\My Documents\Personal" />
<directory name="\My Documents\Business" />
```

```
<directory name="\My Documents\My Music">
<file type="5" name="Alouette" extension="mid" full_path="\My Documents\My
    Music\Alouette.mid" size="13561" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="Alouette" count="1" />
<word name="mid" count="1" />
  </file>
<file type="5" name="ding" extension="amr" full_path="\My Documents\My
    Music\ding.amr" size="1248" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="ding" count="1" />
<word name="amr" count="1" />
  </file>
<file type="5" name="Dogbark" extension="wav" full_path="\My Documents\My
    Music\Dogbark.wav" size="42340" last_access_date="8/21/06"
    last_access_time="12:00:00 AM">
<word name="Dogbark" count="1" />
<word name="wav" count="1" />
  </file>
  </directory>
<directory name="\My Documents\UAContents">
<file type="5" name="CannedText" extension="txt" full_path="\My
    Documents\UAContents\CannedText.txt" size="1215"
    last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="CannedText" count="1" />
<word name="txt" count="1" />
<word name="  Call" count="1" />
<word name="me  Can't" count="1" />
<word name="wait" count="1" />
<word name="to" count="1" />
<word name="see" count="1" />
<word name="you Chill" count="1" />
<word name="out  Congratulations!" count="1" />
<word name="Crazy" count="1" />
<word name="For" count="1" />
<word name="You  Did" count="1" />
<word name="you" count="3" />
<word name="forget?  Dream" count="1" />
<word name="OnFun" count="1" />
<word name="in" count="1" />
<word name="the" count="1" />
<word name="Sun" count="1" />
<word name="Get" count="1" />
<word name="a" count="2" />
<word name="Clue" count="1" />
<word name="Good" count="1" />
<word name="Times  Groovy Guess" count="1" />
<word name="Who  Happy" count="1" />
<word name="Anniversary!  Happy" count="1" />
<word name="Birthday!" count="1" />
<word name="I" count="1" />
<word name="Love" count="1" />
```

<word name="You   I'm" count="1" />
<word name="Really" count="1" />
<word name="Sorry Just" count="1" />
<word name="JokingKiss" count="1" />
<word name="Kiss" count="1" />
<word name="KissLet's" count="1" />
<word name="barbeque Let's" count="1" />
<word name="dance   Let's" count="1" />
<word name="get" count="1" />
<word name="together." count="1" />
<word name="Let's" count="1" />
<word name="talk" count="1" />
<word name="Look" count="1" />
<word name="at" count="1" />
<word name="this! Missing" count="1" />
<word name="You   No" count="1" />
<word name="Monkey" count="1" />
<word name="Business   No" count="1" />
<word name="Way" count="1" />
<word name="No" count="1" />
<word name="Worries   Oops! Peace" count="1" />
<word name="Out   Rock" count="1" />
<word name="On! Take" count="1" />
<word name="Hint   Thanks" count="1" />
<word name="for" count="1" />
<word name="being" count="1" />
<word name="there   Thinking" count="1" />
<word name="of" count="1" />
<word name="Top" count="1" />
<word name="Secret   What" count="1" />
<word name="would" count="1" />
<word name="I" count="1" />
<word name="do" count="1" />
<word name="without" count="1" />
<word name="you?   Wish" count="1" />
<word name="were" count="1" />
<word name="here You" count="1" />
<word name="amaze" count="1" />
<word name="me   You" count="1" />
<word name="crack" count="1" />
<word name="me" count="2" />
<word name="up You" count="1" />
<word name="did" count="1" />
<word name="it!   You" count="1" />
<word name="drive" count="1" />
<word name="nuts! You" count="1" />
<word name="rock!" count="1" />
</file>
</directory>
<directory name="\My Documents\UAContents\Templates">

<file type="5" name="0" extension="dat" full_path="\My Documents\UAContents\Templates\0.dat" size="37330" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="0" count="1" />
<word name="dat" count="1" />
</file>
<file type="5" name="0" extension="jpg" full_path="\My Documents\UAContents\Templates\0.jpg" size="3631" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="0" count="1" />
<word name="jpg" count="1" />
</file>
<file type="5" name="1" extension="dat" full_path="\My Documents\UAContents\Templates\1.dat" size="37476" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="1" count="1" />
<word name="dat" count="1" />
</file>
<file type="5" name="1" extension="jpg" full_path="\My Documents\UAContents\Templates\1.jpg" size="4491" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="1" count="1" />
<word name="jpg" count="1" />
</file>
<file type="5" name="2" extension="dat" full_path="\My Documents\UAContents\Templates\2.dat" size="40504" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="2" count="1" />
<word name="dat" count="1" />
</file>
<file type="5" name="2" extension="jpg" full_path="\My Documents\UAContents\Templates\2.jpg" size="2776" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="2" count="1" />
<word name="jpg" count="1" />
</file>
<file type="5" name="Custom" extension="jpg" full_path="\My Documents\UAContents\Templates\Custom.jpg" size="1458" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Custom" count="1" />
<word name="jpg" count="1" />
</file>
<file type="5" name="MMSTemplates" extension="inf" full_path="\My Documents\UAContents\Templates\MMSTemplates.inf" size="740" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="MMSTemplates" count="1" />
<word name="inf" count="1" />
</file>
</directory>
<directory name="\My Documents\UAContents\MMS UA" />
<directory name="\My Documents\Windows" />
<directory name="\Program Files">

```
<file type="5" name="Mobile PIM Master" extension="exe" full_path="\Program Files\Mobile PIM Master.exe" size="1145344" last_access_date="8/6/07" last_access_time="12:00:00 AM">
<word name="Mobile PIM Master" count="1" />
<word name="exe" count="1" />
  </file>
  </directory>
<directory name="\Program Files\Connections" />
<directory name="\Program Files\Windows Media Player">
<file type="5" name="Default(Portrait)" extension="skn" full_path="\Program Files\Windows Media Player\Default(Portrait).skn" size="26" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Default(Portrait)" count="1" />
<word name="skn" count="1" />
  </file>
<file type="5" name="Default(Landscape)" extension="skn" full_path="\Program Files\Windows Media Player\Default(Landscape).skn" size="27" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Default(Landscape)" count="1" />
<word name="skn" count="1" />
  </file>
<file type="5" name="Default(Square)" extension="skn" full_path="\Program Files\Windows Media Player\Default(Square).skn" size="24" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Default(Square)" count="1" />
<word name="skn" count="1" />
  </file>
  </directory>
<directory name="\Program Files\Microsoft Pocket Streets">
<file type="5" name="PSPimInt" extension="dll" full_path="\Program Files\Microsoft Pocket Streets\PSPimInt.dll" size="24624" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="PSPimInt" count="1" />
<word name="dll" count="1" />
  </file>
<file type="5" name="Pst110" extension="dll" full_path="\Program Files\Microsoft Pocket Streets\Pst110.dll" size="51760" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Pst110" count="1" />
<word name="dll" count="1" />
  </file>
<file type="5" name="Pstreets" extension="exe" full_path="\Program Files\Microsoft Pocket Streets\Pstreets.exe" size="391728" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="Pstreets" count="1" />
<word name="exe" count="1" />
  </file>
  </directory>
<directory name="\Application Data" />
<directory name="\Application Data\Volatile">
```

```
<file type="5" name="setup" extension="log" full_path="\Application Data\Volatile\setup.log" size="2268" last_access_date="8/6/07" last_access_time="12:00:00 AM">
<word name="setup" count="1" />
<word name="log" count="1" />
  </file>
  </directory>
<directory name="\Application Data\Volatile\XMEInst000304A0-33DF273E" />
<directory name="\Application Data\Sounds" />
<directory name="\ConnMgr">
<file type="5" name="" extension="" full_path="\ConnMgr\CMMapG" size="80" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="" count="2" />
  </file>
<file type="5" name="" extension="" full_path="\ConnMgr\CMMapP" size="78" last_access_date="8/21/06" last_access_time="12:00:00 AM">
<word name="" count="2" />
  </file>
  </directory>
  </directories>
  </index>
```

LIST OF REFERENCES

[1] D. Barreau, “Context as a factor in personal information management systems,” *Journal of the American Society for Information Science*, vol. 46, issue 5, pp. 327-339, June 1995.

[2] V. Bush, “As we may think,” *The Atlantic Monthly*, pp. 101-108, July 1945.

[3] G.A. Miller, “Psychology and information,” *American Documentation*, July 1968, pp. 286-289.

[4] T. W. Malone, “How do people organize their desks?: Implications for the design of office information systems,” *ACM Transitions on Information Systems*, vol. 1, issue 1, pp. 99-112, January 1983.

[5] D. Barreau and B. A. Nardi, “Finding and reminding: file organization from the desktop,” *SIGCHI Bulletin*, vol. 27, No. 3, pp. 39-43, July 1995.

[6] B. Shneiderman, D. Byrd and W. B. Croft, “Sorting out searching: a user-interface framework for text searches,” *Communications of the ACM*, vol. 41 No. 4, pp. 95-98, April 1998.

[7] S. Dumais, E. Cutrell, J. Cadiz, G. Jancke,, R. Sarin, and D. C. Robbins, “Stuff I've seen: a system for personal information retrieval and re-use,” in *Proceedings of the 26th Annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003, pp. 72-79.

[8] R. Boardman and M. A. Sasse, "Stuff goes into the computer and doesn't come out: a cross-tool study of personal information management,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2004, pp. 583-590.

[9] J. A. Chirita, S. Costache, J. Gaugaz, and W. Nejdl, “Desktop context detection using implicit feedback,” in *Special Interest Group for Information Retrieval Workshop*, 2006, pp. 24-27.

[10] V. Bellotti, B. Dalal, N. Good, P. Flynn, D. G. Bobrow, and N. Ducheneaut, “What a to-do: studies of task management towards the design of a personal task list manager,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing System*, 2004, pp. 735-742.

[11] V. Bellotti and I. Smith, “Informing the design of an information management system with iterative fieldwork,” in *Proceedings of the Conference on Designing interactive Systems: Processes, Practices, Methods, and Techniques*, 2000, pp. 227-237.

[12] M. Cignini, S. Mizzaro, C. Tasso, and A. Virgili, “E-mail on the move: categorization, filtering, and alerting on mobile devices with the ifMail prototype,” in *Mobile and Ubiquitous Information Access Workshop*, 2003, pp. 107-123.

[13] M. Doncheva, S. M. Drucker, G. Wade, D. Salesin, and M. F. Cohen, “Collecting and organizing web content,” in *Personal Information Management - Special Interest Group for Information Retrieval Workshop*, 2006, pp. 44-47.

[14] E. Cutrell, D. C. Robbins, S. T. Dumais, and R. Sarin, “Fast, flexible filtering with phlat – personal searcg and organization made easy,” in *Computer Human Interaction Proceedings - Personal Information Management*, 2006, pp. 261-270.

[15] M. Czerwinski, “From scatterbrained to focused: UI support for today’s crazed information worker,” *Personal Information Management - Special Interest Group for Information Retrieval Workshop*, 2006, pp. 36-39.

[16] G. Singh, “PIM for mobility,” *Personal Information Management - Special Interest Group for Information Retrieval Workshop*, 2006, pp. 98-101.

[17] H. Shimakawa, S. Kaede, E. Nakamura and S. Azuma, “Weaving threads of action for personal information management,” *International Symposium on Communications and Information Technologies*, 2004, pp. 787-792.

[18] M. Akbas, "Personal information search on mobile devices," in Proceedings of the International Conference on Mobile Technology, Applications and Systems, Singapore, 2007 (in press).

[19] P. J. Denning, “Mastering the mess,” *Communications of the ACM*, vol. 50, issue 4, pp. 21-25, April 2007.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Fort Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marine Corps Representative
   Naval Postgraduate School
   Monterey, California

4. Director, Training and Education, MCCDC, Code C46
   Quantico, Virginia

5. Director, Marine Corps Research Center, MCCDC, Code C40RC
   Quantico, Virginia

6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
   Camp Pendleton, California

7. Head, Information Operations and Space Integration Branch, PLI/PP&O/HQMC, Washington, DC